Inspection des protocoles Internet de base

Les rubriques suivantes expliquent l'inspection des applications pour les protocoles Internet de base. Pour en savoir plus sur la raison pour laquelle vous devez utiliser l'inspection pour certains protocoles et sur les méthodes globales d'application de l'inspection, consultez Premiers pas avec l'inspection du protocole de la couche application.

Inspection DCERPC

L'inspection DCERPC n'est pas activée dans la politique d'inspection par défaut, vous devez donc l'activer si vous avez besoin de cette inspection. Vous pouvez simplement modifier la politique d'inspection globale par défaut pour ajouter l'inspection DCERPC. Vous pouvez également créer une nouvelle politique de service comme vous le souhaitez, par exemple, une politique spécifique à l'interface.

Les sections suivantes décrivent le moteur d'inspection DCERPC.

Présentation de DCERPC

L'appel de procédure à distance de Microsoft (MSRPC), basé sur DCERPC, est un protocole très utilisé par les applications client et serveur distribuées de Microsoft qui permet aux clients logiciels d'exécuter des programmes sur un serveur à distance.

Cela implique généralement qu'un client interroge un serveur appelé mappeur de terminaux à l'écoute sur un numéro de port bien connu pour obtenir les informations réseau allouées dynamiquement d'un service requis. Le client établit ensuite une connexion secondaire avec l'instance de serveur qui fournit le service. Le périphérique de sécurité autorise le numéro de port et l'adresse réseau appropriés et applique également la NAT, le cas échéant, pour la connexion secondaire.

Le moteur d'inspection DCERPC inspecte la communication TCP native entre l'EPM et le client sur le port TCP 135 bien connu. Les opérations de mappage et de recherche d'EPM sont prises en charge pour les clients. Le client et le serveur peuvent se trouver dans n'importe quelle zone de sécurité. L'adresse IP du serveur intégré et le numéro de port sont reçus à partir des messages de réponse EPM applicables. Étant donné qu'un client peut tenter plusieurs connexions au port de serveur renvoyé par EPM, l'utilisation multiple de passages est autorisée, avec des délais d'expiration configurables.

L'inspection DCE prend en charge les identifiants universels uniques (UUID) et les messages suivants :

- UUID du mappeur de terminal (EPM). Tous les messages EPM sont pris en charge.
- UUID ISystemMapper (non EPM). Les messages pris en charge sont les suivants :
  - RemoteCreateInstance opnum4
  - RemoteGetClassObject opnum3
- UUID d'OxidResolver (non EPM). Le message pris en charge est :
  - ServerAlive2 opnum5
- Tout message qui ne contient pas d'adresse IP ou d'informations de port, car ces messages ne nécessitent pas d'inspection.

Configurer une liste des politiques d'inspection DCERPC

Pour spécifier des paramètres d'inspection DCERPC supplémentaires, créez une liste des politiques d'inspection DCERPC. Vous pouvez ensuite appliquer la liste des politiques d'inspection lorsque vous activez l'inspection DCERPC.

Lors de la définition des critères de correspondance du trafic, vous pouvez soit créer une carte de trafic, soit inclure les instructions de correspondance directement dans la liste des politiques. La différence entre la création d'une carte de trafic et la définition de la correspondance de trafic directement dans la liste des politiques d'inspection est que vous pouvez réutiliser les cartes de trafic.

Procédure

Étape 1 (Facultatif) Créez une carte de trafic d'inspection DCERPC.

Pour le trafic que vous identifiez dans cette carte de trafic, vous spécifiez les actions à prendre sur le trafic dans la liste des politiques d'inspection.

Si vous souhaitez effectuer différentes actions pour chaque commande **match**, vous devez identifier le trafic directement dans la liste des politiques.

a) Créez la carte de trafic : **class-map type inspect dcerpc** [**match-all** | **match-any**] *class_map_name*

Où *class_map_name* est le nom de la carte de trafic. Le mot-clé **match-all** est le mot-clé par défaut. Il spécifie que le trafic doit correspondre à tous les critères pour correspondre à la carte de trafic. Le mot-clé match-any spécifie que le trafic correspond à la carte de trafic s'il correspond à au moins un énoncé **match**. L'interface de ligne de commande passe en mode de configuration class-map.

b) Précisez le trafic sur lequel vous souhaitez effectuer des actions à l'aide de la commande **match** suivante. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

- **matchuuid** *type*[**not**] : correspond à l'identifiant unique universel (UUID) du message DCERPC. Le *type* peut être l'un des suivants :
  - **ms-rpc-epm**: correspond aux messages EPM Microsoft RPC.
  - **ms-rpc-isystemactivator**: correspond aux messages ISystemMapper.
  - **ms-rpc-oxidresolver**: correspond aux messages OxidResolver.

c) Saisissez **exit** pour quitter le mode de configuration de carte de trafic.

Étape 2 Créer une liste des politiques d'inspection DCERPC : **policy-map type inspect dcerpc** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration policy-map.

Étape 3 (Facultatif) Ajoutez une description à la liste des politiques : **description** *chaîne*

Étape 4 Pour appliquer les actions au trafic correspondant, procédez comme suit.

a) Précisez le trafic sur lequel vous souhaitez effectuer des actions en utilisant l'une des méthodes suivantes :

- Si vous avez créé une carte de trafic DCERPC, spécifiez-la en entrant la commande suivante : **class** *class_map_name*
- Précisez le trafic directement dans la liste des politiques à l'aide de l'une des commandes **match** décrites pour les cartes de trafic DCERPC. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

b) Précisez l'action que vous souhaitez effectuer sur le trafic correspondant en saisissant l'une des commandes suivantes :

- **reset** [**log**] : abandonne le paquet, ferme la connexion et envoie une réinitialisation TCP au serveur ou au client.
- **log** : envoie un message de journal système. Vous pouvez utiliser cette option seule ou avec l'une des autres actions.

Vous pouvez spécifier plusieurs commandes **class** ou **match** dans la carte de politiques.

Exemple :

```
hostname(config)# policy-map type inspect dcerpc dcerpc-map
hostname(config-pmap)# match uuid ms-rpc-epm
hostname(config-pmap-c)# log
```

Étape 5 Pour configurer les paramètres qui influent sur le moteur d'inspection, procédez comme suit :

a) Entrez en mode de configuration des paramètres :

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b) Définissez un ou plusieurs paramètres. Vous pouvez définir les options suivantes; utilisez la forme **no** de la commande pour désactiver l'option :

- **timeout pinhole** *hh:mm:ss* : configure le délai d'expiration pour les passages DCERPC et remplace le délai d'expiration du point d'épingle du système global de deux minutes. Le délai d'expiration peut être compris entre 00:00:01 et 119:00:00.
- **endpoint-mapper** [**epm-service-only**] [**lookup-operation** [**timeout** *hh:mm:ss*]] : configure les options pour le trafic du mappeur de point de terminaison. Le mot-clé **epm-service-only** applique le service de mappage de point de terminaison lors de la liaison afin que seul le trafic de service soit traité. Le mot-clé **lookup-operation** active l'opération de recherche du service de mappage de point de terminaison. Vous pouvez configurer le délai d'expiration des passages générés par l'opération de recherche. Si aucun délai d'expiration n'est configuré pour l'opération de recherche, la commande timeout pinhole ou la valeur par défaut est utilisée.

Exemples

L'exemple suivant montre comment définir une liste des politiques d'inspection DCERPC avec le délai d'expiration configuré pour les passages DCERPC.

```
hostname(config)# policy-map type inspect dcerpc dcerpc_map
hostname(config-pmap)# timeout pinhole 0:10:00

hostname(config)# class-map dcerpc
hostname(config-cmap)# match port tcp eq 135

hostname(config)# policy-map global-policy
hostname(config-pmap)# class dcerpc
hostname(config-pmap-c)# inspect dcerpc dcerpc-map

hostname(config)# service-policy global-policy global
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection DNS

L'inspection DNS est activée par défaut. Vous ne devez le configurer que si vous souhaitez un traitement autre que celui par défaut. Les sections suivantes décrivent l'inspection des applications DNS.

Valeurs par défaut pour l'inspection DNS

L'inspection DNS est activée par défaut, à l'aide de la carte de trafic d'inspection preset_dns_map :

- La longueur maximale du message DNS est de 512 octets.
- L'inspection DNS sur TCP est désactivée.
- La longueur maximale du message DNS du client est automatiquement définie pour correspondre à l'enregistrement de ressource.
- La protection DNS est activée, de sorte que l'ASA supprime la session DNS associée à une requête DNS dès que la réponse DNS est transmise par l'ASA. L'ASA surveille également l'échange de messages pour s'assurer que l'ID de la réponse DNS correspond à l'ID de la requête DNS.
- La traduction de l'enregistrement DNS en fonction de la configuration NAT est activée.
- L'application du protocole est activée, ce qui permet la vérification du format du message DNS, y compris une longueur de nom de domaine d'au plus 255 caractères, une longueur d'étiquette de 63 caractères, la compression et la vérification des pointeurs en boucle.

Consultez les commandes d'inspection DNS par défaut suivantes :

```
class-map inspection_default
 match default-inspection-traffic
policy-map type inspect dns preset_dns_map
 parameters
    message-length maximum client auto
    message-length maximum 512
    dns-guard
    protocol-enforcement
    nat-rewrite
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
! ...
service-policy global_policy global
```

Configurer la liste des politiques d'inspection DNS

Vous pouvez créer une liste des politiques d'inspection DNS pour personnaliser les actions d'inspection DNS si le comportement d'inspection par défaut ne convient pas à votre réseau.

Avant de commencer

Certaines options de correspondance de trafic utilisent des expressions régulières à des fins de correspondance. Si vous avez l'intention d'utiliser l'une de ces techniques, créez d'abord l'expression régulière ou la carte de trafic d'expressions régulières.

Procédure

Étape 1 (Facultatif) Créez une carte de trafic d'inspection DNS en procédant comme suit.

Une carte de trafic regroupe plusieurs correspondances de trafic. Vous pouvez également identifier les commandes **match** directement dans la liste des politiques. La différence entre la création d'une carte de classe et la définition de la correspondance de trafic directement dans la liste des politiques d'inspection est que la carte de classe vous permet de créer des critères de correspondance plus complexes, et vous pouvez réutiliser les cartes de classe.

Pour spécifier le trafic qui ne doit pas correspondre à la carte de trafic, utilisez la commande **match not**. Par exemple, si la commande **match not** spécifie la chaîne « example.com », tout trafic qui inclut « example.com » ne correspond pas à la carte de classes.

Pour le trafic que vous identifiez dans cette carte de trafic, vous spécifiez les actions à prendre sur le trafic dans la liste des politiques d'inspection.

Si vous souhaitez effectuer différentes actions pour chaque commande **match**, vous devez identifier le trafic directement dans la liste des politiques.

a) Créez la carte de trafic : **class-map type inspect dns** [**match-all** | **match-any**] *class_map_name*

Où *class_map_name* est le nom de la carte de trafic. Le mot-clé **match-all** est le mot-clé par défaut. Il spécifie que le trafic doit correspondre à tous les critères pour correspondre à la carte de trafic. Le mot-clé match-any spécifie que le trafic correspond à la carte de trafic s'il correspond à au moins un énoncé **match**. L'interface de ligne de commande passe en mode de configuration class-map, où vous pouvez saisir une ou plusieurs commandes **match**.

b) (Facultatif) : Ajoutez une description à la carte de trafic : **description** *string*

Où *string* est la description de la carte de trafic (jusqu'à 200 caractères).

c) Précisez le trafic pour lequel vous souhaitez effectuer des actions à l'aide de l'une des commandes **match** suivantes. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

- **match** [**not**] **header-flag** [**eq**] {*f_name* [*f_name*...] | *f_value*} : correspond à l'indicateur DNS. L'argument *f_name* est le nom d'indicateur DNS, l'un des suivants : **AA** (réponse faisant autorité), **QR** (requête), **RA** (récursivité disponible), **RD** (récursivité souhaitée), **TC** (troncation). L'argument *f_value* est la valeur 16 bits en hexadécimal commençant par 0x, de 0x0 à 0xffff. Le mot-clé eq spécifie une correspondance exacte (match all) ; sans le mot-clé **eq**, le paquet n'a besoin de correspondre qu'à l'un des en-têtes spécifiés (match any). Par exemple, **match header-flag AA QR**.
- **match** [**not**] **dns-type {eq** {*t_name* | *t_value*} | **range** *t_value1 t_value2*} : correspond au type DNS. L'argument *t_name* est le nom de type DNS, l'un des suivants : **A** (adresse IPv4), **AXFR** (transfert de zone complet), **CNAME** (nom canonique), **IXFR** (transfert de zone incrémentiel), **NS** (serveur de noms faisant autorité), **SOA** (début d'une zone d'autorité) ou **TSIG** (signature de transaction). Les arguments t_value sont des valeurs quelconques dans le champ de type DNS (0 à 65 535). Le mot-clé range spécifie une plage et le mot-clé eq spécifie une correspondance exacte. Par exemple : **match dns-type eq A**.
- **match** [**not**] **dns-class {eq** {**in** | *c_value*} | **range** *c_value1 c_value2*} : correspond à la classe DNS. La carte est soit **in** (pour Internet) ou c_value, une valeur quelconque de 0 à 65 535 dans le champ

de carte DNS. Le mot-clé range spécifie une plage et le mot-clé eq spécifie une correspondance exacte. Par exemple : **match dns-class eq in**.

- **match** [**not**] {**question** | **resource-record** {**answer** | **authority** | **additional**}} : correspond à une question DNS ou à un enregistrement de ressource. Le mot-clé **question** spécifie la partie question d'un message DNS. Le mot-clé resource-record spécifie l'une de ces sections de l'enregistrement de ressource : **answer**, authority, ou **additional**. Par exemple : **match resource-record answer**.
- **match** [**not**] **domain-name regex** {*regex_name* | **class** *class_name*} : correspond à la liste de noms de domaine du message DNS avec l'expression régulière ou la liste d'expressions régulières spécifiée.

d) Saisissez **exit** pour quitter le mode de configuration class-map.

Étape 2 Créer une liste des politiques d'inspection DNS : **policy-map type inspect dns** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration policy-map.

Étape 3 (Facultatif) Ajoutez une description à la liste des politiques : **description** *string*

Étape 4 Pour appliquer les actions au trafic correspondant, procédez comme suit.

a) Précisez le trafic sur lequel vous souhaitez effectuer des actions en utilisant l'une des méthodes suivantes :

- Si vous avez créé une carte de trafic DNS, spécifiez-la en entrant la commande suivante : **class** *class_map_name*
- Précisez le trafic directement dans la liste des politiques à l'aide de l'une des commandes **match** décrites pour les cartes de trafic DNS. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

b) Précisez l'action que vous souhaitez effectuer sur le trafic correspondant en saisissant l'une des commandes suivantes :

- **drop** [**log**] : abandonne tous les paquets qui correspondent.
- **drop-connection** [**log**] : abandonne le paquet et ferme la connexion.
- **mask** [**log**] : masque la partie correspondante du paquet. Cette action est disponible uniquement pour les correspondances d'indicateurs d'en-tête.
- **log** : envoie un message de journal système. Vous pouvez utiliser cette option seule ou avec l'une des autres actions.
- **enforce-tsig** [**drop**] [**log**] : applique la présence de l'enregistrement de ressource TSIG dans un message. Vous pouvez abandonner un paquet sans l'enregistrement de ressource TSIG, le journaliser, ou l'abandonner et le journaliser. Vous pouvez utiliser cette option en conjonction avec l'action de masque pour les correspondances d'indicateurs d'en-tête ; dans le cas contraire, cette action est exclusive avec les autres actions.

Vous pouvez spécifier plusieurs commandes **class** ou **match** dans la liste des politiques. Pour en savoir plus sur l'ordre des commandes **class** et **match**, consultez Comment plusieurs classes de trafic sont gérées.

Exemple :

```
hostname(config)# policy-map type inspect dns dns-map
hostname(config-pmap)# class dns-class-map
hostname(config-pmap-c)# drop
```

```
hostname(config-pmap-c)# match header-flag eq aa
hostname(config-pmap-c)# drop log
```

Étape 5 Pour configurer les paramètres qui influent sur le moteur d'inspection, procédez comme suit :

a) Entrez dans le mode de configuration des paramètres.

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b) Définissez un ou plusieurs paramètres. Vous pouvez définir les options suivantes; utilisez la forme **no** de la commande pour désactiver l'option.

- **dnscrypt** : active DNScrypt pour chiffrer les connexions entre le périphérique et Cisco Umbrella. L'activation de DNScrypt démarre le fil d'échange de clés avec le résolveur Umbrella. Le fil d'échange de clés effectue l'établissement de liaison avec le résolveur toutes les heures et met à jour le périphérique avec une nouvelle clé secrète. Comme DNScrypt utilise UDP/443, vous devez vous assurer que la carte de trafic utilisée pour l'inspection DNS comprend ce port. Notez que la classe d'inspection par défaut comprend déjà UDP/443 pour l'inspection DNS.
- **dns-guard** : active DNS Guard. L'ASA supprime la session DNS associée à une requête DNS dès que la réponse DNS est transmise par l'ASA. L'ASA surveille également l'échange de messages pour s'assurer que l'ID de la réponse DNS correspond à l'ID de la requête DNS.
- **id-mismatch count** *number* **duration** *seconds* **action log** : active la journalisation pour les non-concordances d'ID DNS excessives, où les arguments **count** *number* **duration** *seconds* spécifient le nombre maximal d'instances de non-concordance par seconde avant l'envoi d'un journal de messages système.
- **id-randomization** : randomise l'identifiant DNS pour une requête DNS.
- **message-length maximum** {*length* | **client** {*length* | **auto**} | **server** {*length* | **auto**}} : définit la longueur maximale du message DNS, de 512 à 65 535 octets. Vous pouvez également définir la longueur maximale des messages client ou serveur. Le mot-clé **auto** définit la longueur maximale selon la valeur indiquée dans l'enregistrement de ressource.
- **nat-rewrite** : traduit l'enregistrement DNS en fonction de la configuration NAT.
- **protocol-enforcement** : active la vérification du format du message DNS, y compris la longueur du nom de domaine de 255 caractères maximum, la longueur d'étiquette de 63 caractères, la compression et la vérification des pointeurs en boucle.
- **tcp-inspection** : active l'inspection du trafic DNS sur TCP. Assurez-vous que le trafic DNS/TCP du port 53 fait partie de la carte de trafic à laquelle vous appliquez l'inspection DNS. La carte de trafic d'inspection par défaut comprend TCP/53.
- **tsig enforced action** {[**drop**] [**log**]} : nécessite la présence d'un enregistrement de ressource TSIG. Vous pouvez **abandonner** un paquet non conforme, **journaliser** le paquet ou les deux.
- **umbrella** [**tag** *Umbrella_policy*] [**fail-open**] : active Cisco Umbrella et spécifie éventuellement le nom de la politique Cisco Umbrella (**tag**) à appliquer au périphérique. Si vous ne spécifiez pas de politique, la politique par défaut est appliquée. Pour en savoir plus, consultez Cisco Umbrella.

  Incluez le mot-clé **fail-open** si vous souhaitez que la résolution DNS fonctionne si le serveur DNS Cisco Umbrella n'est pas disponible. En cas d'échec en mode ouvert, si le serveur DNS Cisco Umbrella n'est pas disponible, Umbrella se désactive dans cette liste des politiques et permet aux

demandes DNS d'accéder aux autres serveurs DNS configurés sur le système, le cas échéant. Lorsque les serveurs DNS Umbrella sont de nouveau disponibles, la liste des politiques reprend leur utilisation. Si vous n'incluez pas cette option, les requêtes DNS continuent d'être acheminées au résolveur Cisco Umbrella inaccessible, de sorte qu'elles ne recevront pas de réponse.

Exemple :

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)# dns-guard
hostname(config-pmap-p)# message-length maximum 1024
hostname(config-pmap-p)# nat-rewrite
hostname(config-pmap-p)# protocol-enforcement
```

Exemple

L'exemple suivant montre comment utiliser une nouvelle carte de politiques d'inspection dans la configuration globale par défaut :

```
regex domain_example "example\.com"
regex domain_foo "foo\.com"

! define the domain names that the server serves
class-map type inspect regex match-any my_domains
   match regex domain_example
   match regex domain_foo

! Define a DNS map for query only
class-map type inspect dns match-all pub_server_map
   match not header-flag QR
   match question
   match not domain-name regex class my_domains

policy-map type inspect dns new_dns_map
    class pub_server_map
        drop log
    match header-flag RD
        mask log
    parameters
        message-length maximum client auto
        message-length maximum 512
        dns-guard
        protocol-enforcement
        nat-rewrite

policy-map global_policy
 class inspection_default
  no inspect dns preset_dns_map
  inspect dns new_dns_map
service-policy global_policy global
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection FTP

L'inspection FTP est activée par défaut. Vous ne devez le configurer que si vous souhaitez un traitement autre que celui par défaut. Les sections suivantes décrivent le moteur d'inspection FTP.

Présentation de l'inspection FTP

L'inspection d'application FTP inspecte les sessions FTP et effectue quatre tâches :

- Prépare les canaux de connexion de données secondaires dynamiques pour le transfert de données FTP. Les ports pour ces canaux sont négociés au moyen de commandes PORT ou PASV. Les canaux sont alloués en réponse à un chargement de fichier, à un téléchargement de fichier ou à un événement de liste de répertoires
- Effectue le suivi de la séquence de commande-réponse FTP.
- Génère une trace d'audit.
  - L'enregistrement d'audit 303002 est généré pour chaque fichier récupéré ou chargé.
  - L'enregistrement d'audit 201005 est généré si la préparation du canal dynamique secondaire a échoué en raison d'un manque de mémoire.
- Traduit l'adresse IP intégrée.

Remarque Si vous désactivez l'inspection FTP, les utilisateurs sortants ne peuvent démarrer les connexions qu'en mode passif, et tous les FTP entrants sont désactivés.

FTP strict

Le protocole FTP strict augmente la sécurité des réseaux protégés en empêchant les navigateurs Web d'envoyer des commandes intégrées dans les demandes FTP. Pour activer le protocole FTP strict, incluez l'option stricte avec la commande **inspect ftp**.

Lorsque vous utilisez un protocole FTP strict, vous pouvez éventuellement spécifier une liste des politiques d'inspection FTP pour préciser les commandes FTP qui ne sont pas autorisées à passer par l'ASA.

L'inspection FTP strict applique le comportement suivant :

- Une commande FTP doit être reconnue avant que l'ASA autorise une nouvelle commande.
- L'ASA abandonne les connexions qui envoient des commandes intégrées.
- Les commandes 227 et PORT sont vérifiées pour s'assurer qu'elles ne s'affichent pas dans une chaîne d'erreur.

Mise en garde L'utilisation de FTP strict peut entraîner des défaillances chez les clients FTP qui ne sont pas strictement conformes aux RFC FTP. En outre, vous devez vous assurer d'appliquer l'inspection à vos ports FTP uniquement (TCP/21 est le port FTP normal). Une inspection FTP stricte appliquée au trafic non FTP peut entraîner une perte de trafic inattendue, en particulier le trafic HTTP.

Grâce à une inspection FTP stricte, chaque commande et séquence de réponse FTP est suivie pour l'activité anormale suivante :

- Commande tronquée : le nombre de virgules dans la commande de réponse PORT et PASV est vérifié pour voir s'il est de cinq. S'il n'y a pas cinq, la commande PORT est considérée comme tronquée et la connexion TCP est fermée.
- Commande incorrecte : vérifie la commande FTP pour voir si elle se termine par <CR><LF> caractères, comme l'exige la RFC. Si ce n'est pas le cas, la connexion est fermée.
- Taille des commandes RETR et STOR : celles-ci sont vérifiées par rapport à une constante fixe. Si la taille est supérieure, un message d'erreur est enregistré et la connexion est fermée.
- Usurpation de commande : la commande PORT doit toujours être envoyée par le client. La connexion TCP est refusée si une commande PORT est envoyée à partir du serveur.
- Usurpation de réponse : la commande de réponse PASV (227) doit toujours être envoyée à partir du serveur. La connexion TCP est refusée si une commande de réponse PASV est envoyée par le client. Cela empêche le gouffre de sécurité lorsque l'utilisateur exécute « 227 xxxx a1, a2, a3, a4, p1, p2 ».
- Modification du flux TCP : l'ASA ferme la connexion s'il détecte une modification du flux TCP.
- Négociation de port non valide : la valeur de port dynamique négociée est vérifiée pour voir si elle est inférieure à 1024. Comme les numéros de port dans la plage de 1 à 1024 sont réservés aux connexions bien connues, si le port négocié se trouve dans cette plage, la connexion TCP est libérée.
- Filtrage de commande : le nombre de caractères présents après les numéros de port dans les commandes de réponse PORT et PASV est vérifié par recoupement avec une valeur constante de 8. S'il est supérieur à 8, la connexion TCP est fermée.
- L'ASA remplace la réponse du serveur FTP à la commande SYST par une série de X pour empêcher le serveur de révéler son type de système aux clients FTP. Pour remplacer ce comportement par défaut, utilisez la commande **no mask-syst-reply** dans la liste FTP.

Configurer une liste des politiques d'inspection FTP

Le filtrage des commandes FTP et les vérifications de sécurité sont fournis à l'aide de l'inspection FTP stricte pour améliorer la sécurité et le contrôle. La conformité au protocole comprend les vérifications de la longueur des paquets, les vérifications des délimiteurs et du format des paquets, les vérifications du terminateur de commande et la validation des commandes.

Le blocage de FTP en fonction des valeurs des utilisateurs est également pris en charge afin que les sites FTP puissent publier des fichiers à télécharger, tout en limitant l'accès à certains utilisateurs. Vous pouvez bloquer les connexions FTP en fonction du type de fichier, du nom du serveur et d'autres attributs. Des journaux de messages système sont générés si une connexion FTP est refusée après l'inspection.

Si vous souhaitez que l'inspection FTP permette aux serveurs FTP de révéler leur type de système aux clients FTP et limite les commandes FTP autorisées, créez et configurez une carte de politique d'inspection FTP. Vous pouvez ensuite appliquer la liste lorsque vous activez l'inspection FTP.

Avant de commencer

Certaines options de correspondance de trafic utilisent des expressions régulières à des fins de correspondance. Si vous avez l'intention d'utiliser l'une de ces techniques, créez d'abord l'expression régulière ou la carte de trafic d'expressions régulières.

Procédure

Étape 1 (Facultatif) Créez une carte de trafic d'inspection FTP en procédant comme suit.

Une carte de trafic regroupe plusieurs correspondances de trafic. Vous pouvez également identifier les commandes **match** directement dans la liste des politiques. La différence entre la création d'une carte de classe et la définition de la correspondance de trafic directement dans la liste des politiques d'inspection est que la carte de classe vous permet de créer des critères de correspondance plus complexes, et vous pouvez réutiliser les cartes de classe.

Pour spécifier le trafic qui ne doit pas correspondre à la carte de trafic, utilisez la commande **match not**. Par exemple, si la commande **match not** spécifie la chaîne « example.com », tout trafic qui inclut « example.com » ne correspond pas à la carte de classes.

Pour le trafic que vous identifiez dans cette carte de trafic, vous spécifiez les actions à prendre sur le trafic dans la liste des politiques d'inspection.

Si vous souhaitez effectuer différentes actions pour chaque commande **match**, vous devez identifier le trafic directement dans la liste des politiques.

a) Créez la carte de trafic : **class-map type inspect ftp** [**match-all** | **match-any**] *class_map_name*

Où *class_map_name* est le nom de la carte de trafic. Le mot-clé **match-all** est le mot-clé par défaut. Il spécifie que le trafic doit correspondre à tous les critères pour correspondre à la carte de trafic. Le mot-clé match-any spécifie que le trafic correspond à la carte de trafic s'il correspond à au moins un énoncé **match**. L'interface de ligne de commande passe en mode de configuration class-map, où vous pouvez saisir une ou plusieurs commandes **match**.

b) (Facultatif) : Ajoutez une description à la carte de trafic : **description** *string*

Où *string* est la description de la carte de trafic (jusqu'à 200 caractères).

c) Précisez le trafic pour lequel vous souhaitez effectuer des actions à l'aide de l'une des commandes **match** suivantes. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

- **match** [**not**] **filename regex** {*regex_name* | **class** *class_name*} : correspond au nom de fichier dans le transfert FTP avec l'expression régulière ou la liste d'expressions régulières spécifiée.
- **match** [**not**] **filetype regex** {*regex_name* | **class** *class_name*} : correspond au type de fichier dans le transfert FTP par rapport à l'expression régulière ou à la liste d'expressions régulières spécifiée.
- **match** [**not**] **request-command** *ftp_command* [*ftp_command*...] correspond à la commande FTP, à l'une ou à plusieurs des commandes suivantes :
  - **APPE** : ajouter à un fichier.

- **CDUP** : passe au répertoire parent du répertoire de travail actuel.
- **DELE** : pour supprimer un fichier sur le serveur.
- **GET** : obtient un fichier du serveur.
- **HELP** : fournit des renseignements d'aide.
- **MKD** : crée un répertoire sur le serveur.
- **PUT** : envoie un fichier au serveur.
- **RMD** : supprime un répertoire sur le serveur.
- **RNFR** : spécifie le nom de fichier « rename-from ».
- **RNTO** : spécifie le nom de fichier « rename-to ».
- **SITE** : utilisé pour spécifier une commande spécifique au serveur. Ceci est généralement utilisé pour l'administration à distance.
- **STOU** : stocke un fichier en utilisant un nom de fichier unique.

- **match** [**not**] **server regex** {*regex_name* | **class** *class_name*} : correspond au nom du serveur FTP avec l'expression régulière ou la liste d'expressions régulières spécifiée.
- **correspondance** [**not**] **nom d'utilisateur regex** {*regex_name* | **class** *class_name*} : correspond au nom d'utilisateur FTP avec l'expression régulière ou la classe d'expression régulière spécifiée.

d) Saisissez **exit** pour quitter le mode de configuration class-map.

Étape 2 Créez une liste des politiques d'inspection FTP : **policy-map type inspect ftp** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration de liste des politiques.

Étape 3 (Facultatif) Ajoutez une description à la liste des politiques : **description** *string*

Étape 4 Pour appliquer les actions au trafic correspondant, procédez comme suit.

a) Précisez le trafic sur lequel vous souhaitez effectuer des actions en utilisant l'une des méthodes suivantes :

- Si vous avez créé une carte de trafic FTP, spécifiez-la en entrant la commande suivante : **class** *class_map_name*
- Précisez le trafic directement dans la liste des politiques à l'aide de l'une des commandes **match** décrites pour les cartes de trafic FTP. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

b) Précisez l'action que vous souhaitez effectuer sur le trafic correspondant en entrant la commande suivante :

- **reset** [**log**] : abandon du paquet, fermeture de la connexion et envoi d'une réinitialisation TCP au serveur ou au client. Ajoutez le mot-clé **log** pour envoyer un message de journal système.

Vous pouvez spécifier plusieurs commandes **class** ou **match** dans la liste des politiques. Pour en savoir plus sur l'ordre des commandes **class** et **match**, consultez Comment plusieurs classes de trafic sont gérées.

Étape 5 Pour configurer les paramètres qui influent sur le moteur d'inspection, procédez comme suit :

a) Entrez dans le mode de configuration des paramètres.

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b) Définissez un ou plusieurs paramètres. Vous pouvez définir les options suivantes; utilisez la forme **no** de la commande pour désactiver l'option :

- **mask-banner** : masque la bannière de message d'accueil à partir du serveur FTP.
- **masque-syst-reply** : masque la réponse à la commande **syst**.

Exemple

Avant de soumettre un nom d'utilisateur et un mot de passe, tous les utilisateurs FTP reçoivent une bannière de message d'accueil. Par défaut, cette bannière comprend des informations de version utiles aux pirates qui tentent d'identifier les faiblesses d'un système. L'exemple suivant montre comment masquer cette bannière :

```
hostname(config)# policy-map type inspect ftp mymap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# mask-banner

hostname(config)# class-map match-all ftp-traffic
hostname(config-cmap)# match port tcp eq ftp

hostname(config)# policy-map ftp-policy
hostname(config-pmap)# class ftp-traffic
hostname(config-pmap-c)# inspect ftp strict mymap

hostname(config)# service-policy ftp-policy interface inside
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection HTTP

L'inspection HTTP n'est pas activée dans la politique d'inspection par défaut, vous devez donc l'activer si vous avez besoin de cette inspection. Cependant, la carte de trafic d'inspection par défaut comprend les ports HTTP par défaut, vous pouvez donc simplement modifier la politique d'inspection globale par défaut pour ajouter l'inspection HTTP. Vous pouvez également créer une nouvelle politique de service comme vous le souhaitez, par exemple, une politique spécifique à l'interface.

Les sections suivantes décrivent le moteur d'inspection HTTP.

Présentation de l'inspection HTTP

Utilisez le moteur d'inspection HTTP pour une protection contre les attaques spécifiques et d'autres menaces associées au trafic HTTP.

L'inspection des applications HTTP analyse les en-têtes et le corps HTTP et effectue diverses vérifications sur les données. Ces vérifications empêchent diverses constructions HTTP, types de contenu et protocoles de tunnellisation et de messagerie de traverser l'appareil de sécurité.

La fonctionnalité d'inspection HTTP améliorée, également connue sous le nom de pare-feu d'application et disponible lorsque vous configurez une liste des politiques d'inspection HTTP, peut aider à empêcher les attaquants d'utiliser les messages HTTP pour contourner la politique de sécurité du réseau.

L'inspection des applications HTTP peut bloquer les applications tunnellisées et les caractères non ASCII dans les demandes et réponses HTTP, ce qui empêche le contenu malveillant d'atteindre le serveur Web. La limitation de taille de divers éléments dans les en-têtes de demande et de réponse HTTP, le blocage d'URL et l'usurpation d'en-tête de serveur HTTP sont également pris en charge.

L'inspection HTTP améliorée vérifie les éléments suivants pour tous les messages HTTP :

- Conformité à la RFC 2616
- Utilisation uniquement des méthodes définies par la RFC.
- Conformité avec les critères supplémentaires.

Configurer une liste des politiques d'inspection HTTP

Pour préciser les actions lorsqu'un message enfreint un paramètre, créez une liste des politiques d'inspection HTTP. Vous pouvez ensuite appliquer la liste des politiques d'inspection lorsque vous activez l'inspection HTTP.

Avant de commencer

Certaines options de correspondance de trafic utilisent des expressions régulières à des fins de correspondance. Si vous avez l'intention d'utiliser l'une de ces techniques, créez d'abord l'expression régulière ou la carte de classe d'expression régulière.

Procédure

Étape 1 (Facultatif) Créez une carte de trafic d'inspection HTTP en procédant comme suit.

Une carte de trafic regroupe plusieurs correspondances de trafic. Vous pouvez également identifier les commandes **match** directement dans la liste des politiques. La différence entre la création d'une carte de classe et la définition de la correspondance de trafic directement dans la liste des politiques d'inspection est que la carte de classe vous permet de créer des critères de correspondance plus complexes, et vous pouvez réutiliser les cartes de classe.

Pour spécifier le trafic qui ne doit pas correspondre à la carte de classe, utilisez la commande **match not**. Par exemple, si la commande **match not** spécifie la chaîne « example.com », tout trafic qui inclut « example.com » ne correspond pas à la carte de trafic.

Pour le trafic que vous identifiez dans cette carte de trafic, vous spécifiez les actions à prendre sur le trafic dans la liste des politiques d'inspection.

Si vous souhaitez effectuer différentes actions pour chaque commande **match**, vous devez identifier le trafic directement dans la liste des politiques.

a) Créez la carte de trafic : **class-map type inspect http** [**match-all** | **match-any**] *class_map_name*

Où *class_map_name* est le nom de la carte de trafic. Le mot-clé **match-all** est le mot-clé par défaut. Il spécifie que le trafic doit correspondre à tous les critères pour correspondre à la carte de trafic. Le mot-clé match-any spécifie que le trafic correspond à la carte de trafic s'il correspond à au moins un énoncé **match**. L'interface de ligne de commande passe en mode de configuration class-map, où vous pouvez saisir une ou plusieurs commandes **match**.

b) (Facultatif) : Ajoutez une description à la carte de trafic : **description** *string*

Où *string* est la description de la carte de trafic (jusqu'à 200 caractères).

c) Précisez le trafic pour lequel vous souhaitez effectuer des actions à l'aide de l'une des commandes **match** suivantes. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

- **match** [**not**] **req-resp content-type mismatch** : correspond au trafic avec un champ content-type dans la réponse HTTP qui ne correspond pas au champ accept dans le message de demande HTTP correspondant.
- **match** [**not**] **request args regex** {*regex_name* | **class** *class_name*} : correspond au texte trouvé dans les arguments du message de demande HTTP par rapport à l'expression régulière ou à la liste d'expressions régulières spécifiée.
- **match** [**not**] **request body** {**regex** {*regex_name* | **class** *class_name*} | **length gt** *bytes*} : correspond au texte trouvé dans le corps du message de demande HTTP avec l'expression régulière ou la liste d'expressions régulières spécifiée, ou aux messages dans lesquels le corps de demande est supérieur à la longueur spécifiée.
- **match** [**not**] **request header** {*field* | **regex** *regex_name*} **regex** {*regex_name* | **class** *class_name*} : correspond au contenu d'un champ dans l'en-tête du message de demande HTTP avec l'expression régulière ou la liste d'expressions régulières spécifiée. Vous pouvez spécifier le nom du champ explicitement ou faire correspondre le nom du champ à une expression régulière. Les noms des champs sont : accept, accept-charset, accept-encoding, accept-language, allow, authorization, cache-control, connection, content-encoding, content-language, content-length, content-location, content-md5, content-range, content-type, cookie, date, expect, expires, from, host, if-match, if-modified-since, if-none-match, if-range, if-unmodified-since, last-modified, max-forwards, pragma, proxy-authorization, range, referer, te, trailer, transfer-encoding, upgrade, user-agent, via, warning.
- **match** [**not**] **request header** {*field* | **regex** {*regex_name* | **class** *class_name*}} {**length gt** *bytes* | **count gt** *number*} : correspond à la longueur des champs spécifiés dans l'en-tête du message de demande HTTP, ou au nombre total de champs (count) dans l'en-tête. Vous pouvez spécifier le nom du champ explicitement ou faire correspondre le nom de champ à une expression régulière ou à une liste d'expressions régulières. Les noms de champs sont répertoriés dans la puce précédente.
- **match** [**not**] **request header** {**length gt** *bytes* | **count gt** *number* | **non-ascii**} : correspond à la longueur totale de l'en-tête du message de demande HTTP, ou au nombre total de champs (count) dans l'en-tête, ou aux en-têtes qui ont des caractères non ASCII.
- **match** [**not**] **request method** {*method* | **regex** {*regex_name* | **class** *class_name*}} : correspond à la méthode de demande HTTP. Vous pouvez spécifier la méthode explicitement ou la faire correspondre

à une expression régulière ou à une liste d'expressions régulières. Les méthodes sont : bcopy, bdelete, bmove, bpropfind, bproppatch, connect, copy, delete, edit, get, getattribute, getattributenames, getproperties, head, index, lock, mkcol, mkdir, move, notify, options, poll, post, propfind, proppatch, put, revadd, revlabel, revlog, revnum, save, search, setattribute, startrev, stoprev, subscribe, trace, unedit, unlock, unsubscribe.

- **match** [**not**] **request uri** {**regex** {*regex_name* | **class** *class_name*} | **length gt** *bytes*} : correspond au texte trouvé dans l'URI du message de demande HTTP avec l'expression régulière ou la liste d'expressions régulières spécifiée, ou aux messages dans lesquels l'URI de demande est supérieur à la longueur spécifiée.
- **match** [**not**] **response body** {**active-x** | **java-applet** | **regex** {*regex_name* | **class** *class_name*}} : correspond au texte trouvé dans le corps du message de réponse HTTP par rapport à l'expression régulière ou à la liste d'expressions régulières spécifiée, ou commente les balises d'applet Java et d'objet ActiveX afin de les filtrer.
- **match** [**not**] **response body length gt** *bytes* : correspond aux messages de réponse HTTP dans lesquels le corps est supérieur à la longueur spécifiée.
- **match** [**not**] **response header** {*field* | **regex** *regex_name*} **regex** {*regex_name* | **class** *class_name*} : correspond au contenu d'un champ dans l'en-tête du message de réponse HTTP avec l'expression régulière ou la liste d'expressions régulières spécifiée. Vous pouvez spécifier le nom du champ explicitement ou faire correspondre le nom du champ à une expression régulière. Les noms des champs sont les suivants : accept-ranges, age, allow, cache-control, connection, content-encoding, content-language, content-length, content-location, content-md5, content-range, content-type, date, etag, expires, last-modified, location, pragma, proxy-authenticate, retry-after, server, set-cookie, trailer, transfer-encoding, upgrade, vary, via, warning, www-authenticate.
- **match** [**not**] **response header** {*field* | **regex** {*regex_name* | **class** *class_name*}} {**length gt** *bytes* | **count gt** *number*} : correspond à la longueur des champs spécifiés dans l'en-tête du message de réponse HTTP ou au nombre total de champs (count) dans l'en-tête. Vous pouvez spécifier le nom du champ explicitement ou faire correspondre le nom de champ à une expression régulière ou à une liste d'expressions régulières. Les noms de champs sont répertoriés dans la puce précédente.
- **match** [**not**] **response header** {**length gt** *bytes* | **count gt** *number* | **non-ascii**} : correspond à la longueur totale de l'en-tête du message de réponse HTTP, ou au nombre total de champs (count) dans l'en-tête, ou aux en-têtes qui ont des caractères non ASCII.
- **match** [**not**] **response status-line regex** {*regex_name* | **class** *class_name*} : correspond au texte trouvé dans la ligne d'état du message de réponse HTTP par rapport à l'expression régulière ou à la classe d'expressions régulières spécifiée.

d) Saisissez **exit** pour quitter le mode de configuration de carte de trafic.

Étape 2 Créez une liste des politiques d'inspection HTTP : **policy-map type inspect http** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration policy-map.

Étape 3 (Facultatif) Ajoutez une description à la liste des politiques : **description** *chaîne*

Étape 4 Pour appliquer les actions au trafic correspondant, procédez comme suit.

a) Précisez le trafic sur lequel vous souhaitez effectuer des actions en utilisant l'une des méthodes suivantes :

- Si vous avez créé une carte de trafic HTTP, spécifiez-la en entrant la commande suivante : **class** *class_map_name*

- Précisez le trafic directement dans la liste des politiques à l'aide de l'une des commandes **match** décrites pour les cartes de trafic HTTP. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

b) Précisez l'action que vous souhaitez effectuer sur le trafic correspondant en saisissant l'une des commandes suivantes :

- **drop-connection** [**log**] : abandonner le paquet et fermer la connexion.
- **reset** [**log**] : abandonne le paquet, ferme la connexion et envoie une réinitialisation TCP au serveur ou au client.
- **log** : envoie un message de journal système. Vous pouvez utiliser cette option seule ou avec l'une des autres actions.

Vous pouvez spécifier plusieurs commandes **class** ou **match** dans la carte de politiques. Pour en savoir plus sur l'ordre des commandes **class** et **match**, consultez Comment plusieurs classes de trafic sont gérées.

Étape 5 Pour configurer les paramètres qui influent sur le moteur d'inspection, procédez comme suit :

a) Entrez en mode de configuration des paramètres :

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b) Définissez un ou plusieurs paramètres. Vous pouvez définir les options suivantes; utilisez la forme **no** de la commande pour désactiver l'option :

- **body-match-maximum** *number* : définit le nombre maximal de caractères dans le corps d'un message HTTP qui doivent être recherchés dans une correspondance de corps. Par défaut, c'est de 200 octets. Un grand nombre aura un impact considérable sur les performances.
- **protocol-violation action** {**drop-connection** [**log**] | **reset** [**log**] | **log**} : vérifie les violations du protocole HTTP. Vous devez également choisir l'action à prendre en cas de violations (abandon de la connexion, réinitialisation ou journalisation) et activer ou désactiver la journalisation.
- **spoof-server** *string* : remplace le champ d'en-tête Server par une chaîne. Les flux WebVPN ne sont pas soumis à la commande spoof-server.

Exemple

L'exemple suivant montre comment définir une liste des politiques d'inspection HTTP qui autorisera et journalisera toute connexion HTTP qui tente d'accéder à « www\.xyz.com/.*\.asp » ou « www\.xyz[0-9][0-9]\.com » avec les méthodes « GET » ou « PUT ». Toutes les autres combinaisons d'URL/méthode seront autorisées silencieusement.

```
hostname(config)# regex url1 "www\.xyz.com/.*\.asp"
hostname(config)# regex url2 "www\.xyz[0-9][0-9]\.com"
hostname(config)# regex get "GET"
hostname(config)# regex put "PUT"

hostname(config)# class-map type regex match-any url_to_log
```

```
hostname(config-cmap)# match regex url1
hostname(config-cmap)# match regex url2
hostname(config-cmap)# exit

hostname(config)# class-map type regex match-any methods_to_log
hostname(config-cmap)# match regex get
hostname(config-cmap)# match regex put
hostname(config-cmap)# exit

hostname(config)# class-map type inspect http http_url_policy
hostname(config-cmap)# match request uri regex class url_to_log
hostname(config-cmap)# match request method regex class methods_to_log
hostname(config-cmap)# exit

hostname(config)# policy-map type inspect http http_policy
hostname(config-pmap)# class http_url_policy
hostname(config-pmap-c)# log
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection ICMP

Le moteur d'inspection ICMP permet au trafic ICMP d'avoir une « session » afin qu'il puisse être inspecté comme le trafic TCP et UDP. Sans le moteur d'inspection ICMP, nous vous recommandons de ne pas autoriser ICMP par l'intermédiaire de l'ASA dans une liste de contrôle d'accès. Sans inspection dynamique, ICMP peut être utilisé pour attaquer votre réseau. Le moteur d'inspection ICMP garantit qu'il n'y a qu'une seule réponse à chaque demande et que le numéro de séquence est correct.

Cependant, le trafic ICMP dirigé vers une interface ASA n'est jamais inspecté, même si vous activez l'inspection ICMP. Ainsi, un ping (requête d'écho) à une interface peut échouer dans des circonstances spécifiques, comme lorsque la demande d'écho provient d'une source que l'ASA peut atteindre par une route de sauvegarde par défaut.

Remarque La NAT utilise l'inspection ICMP lors de la traduction des paquets, même si vous désactivez l'inspection ICMP.

Pour en savoir plus sur l'activation de l'inspection ICMP, consultez Configurer l'inspection du protocole de couche d'application.

Inspection des erreurs ICMP

Lorsque l'inspection des erreurs ICMP est activée, l'ASA crée des sessions de traduction pour les sauts intermédiaires qui envoient des messages d'erreur ICMP, en fonction de la configuration NAT. L'ASA remplace les adresses IP du paquet par les adresses IP traduites.

Lorsque cette option est désactivée, l'ASA ne crée pas de sessions de traduction pour les nœuds intermédiaires qui génèrent des messages d'erreur ICMP. Les messages d'erreur ICMP générés par les nœuds intermédiaires

entre l'hôte interne et l'ASA atteignent l'hôte externe sans utiliser de ressource NAT supplémentaire. Cela n'est pas souhaité lorsqu'un hôte externe utilise la commande traceroute pour retracer les sauts jusqu'à la destination à l'intérieur de l'ASA. Lorsque l'ASA ne traduit pas les sauts intermédiaires, tous les sauts intermédiaires s'affichent avec l'adresse IP de destination mappée.

Remarque Vous devez toujours activer l'inspection d'erreur ICMP s'il est possible que la NAT soit utilisée sur les paquets ICMP. Comme la NAT utilise automatiquement l'inspection ICMP pour les paquets ICMP, même si l'inspection ICMP est désactivée, l'utilisation de l'adresse de destination mappée comme adresse source peut donner l'impression qu'un analyseur examine votre réseau. Par exemple, sans inspection d'erreur ICMP également activée, si le paquet de demande ECHO a sa destination traduite, lorsqu'il est intégré dans une réponse de délai ICMP dépassé, l'en-tête externe de la demande de délai dépassé utilise la destination traduite comme adresse source. Si vous activez l'inspection d'erreur ICMP, l'adresse source de dépassement sera définie à la valeur correcte.

Pour en savoir plus sur l'activation de l'inspection des erreurs ICMP, consultez Configurer l'inspection du protocole de couche d'application.

Inspection ILS

Le moteur d'inspection Internet Locator Service (ILS) prend en charge la NAT pour les produits Microsoft NetMeeting, SiteServer et Active Directory qui utilisent LDAP pour échanger des informations de répertoire avec un serveur ILS. Vous ne pouvez pas utiliser la PAT avec l'inspection ILS, car seules les adresses IP sont stockées par une base de données LDAP.

Pour les réponses de recherche, lorsque le serveur LDAP est situé à l'extérieur, considérez l'utilisation de la NAT pour permettre aux homologues internes de communiquer localement lorsqu'ils sont enregistrés sur des serveurs LDAP externes. Si vous n'avez pas besoin d'utiliser la NAT, nous vous recommandons de désactiver le moteur d'inspection pour obtenir de meilleures performances.

Une configuration supplémentaire peut être nécessaire lorsque le serveur ILS est situé à l'intérieur de la frontière ASA. Cela nécessiterait une ouverture pour que les clients externes accèdent au serveur LDAP sur le port spécifié, généralement TCP 389.

Remarque Comme le trafic ILS (signalisation d'appel H225) ne se produit que sur le canal UDP secondaire, la connexion TCP est déconnectée après l'intervalle d'inactivité TCP. Par défaut, cet intervalle est de 60 minutes et peut être ajusté à l'aide de la commande TCP **timeout**. Dans ASDM, cela se trouve dans le volet **Configuration > Firewall (Pare-feu) > Advanced (Avancé) > Global Timeouts (Délais d'expiration globaux)**.

L'inspection ILS a les limites suivantes :

- Les demandes de parrainage et les réponses ne sont pas prises en charge.
- Les utilisateurs de plusieurs répertoires ne sont pas unifiés.
- Les utilisateurs uniques ayant plusieurs identités dans plusieurs répertoires ne peuvent pas être reconnus par la NAT.

Pour en savoir plus sur l'activation de l'inspection ILS, consultez Configurer l'inspection du protocole de couche d'application.

Inspection de la messagerie instantanée

Le moteur d'inspection de messagerie instantanée (IM) vous permet de contrôler l'utilisation du réseau d'IM et d'arrêter la fuite de données confidentielles, la propagation des vers et d'autres menaces pour le réseau d'entreprise.

L'inspection IM n'est pas activée dans la politique d'inspection par défaut, vous devez donc l'activer si vous avez besoin de cette inspection. Cependant, la carte de trafic d'inspection par défaut comprend les ports IM par défaut, vous pouvez donc simplement modifier la politique d'inspection globale par défaut pour ajouter l'inspection IM. Vous pouvez également créer une nouvelle politique de service comme vous le souhaitez, par exemple, une politique spécifique à l'interface.

Si vous décidez de mettre en œuvre l'inspection IM, vous pouvez également configurer une carte de politique d'inspection IM pour préciser les actions lorsqu'un message enfreint un paramètre. La procédure suivante explique les mappages de politiques d'inspection d'IM.

Avant de commencer

Certaines options de correspondance de trafic utilisent des expressions régulières à des fins de correspondance. Si vous avez l'intention d'utiliser l'une de ces techniques, créez d'abord l'expression régulière ou la carte de trafic d'expressions régulières.

Procédure

Étape 1 (Facultatif) Créez une carte de trafic d'inspection IM en procédant comme suit.

Une carte de trafic regroupe plusieurs correspondances de trafic. Vous pouvez également identifier les commandes **match** directement dans la liste des politiques. La différence entre la création d'une carte de classe et la définition de la correspondance de trafic directement dans la liste des politiques d'inspection est que la carte de classe vous permet de créer des critères de correspondance plus complexes, et vous pouvez réutiliser les cartes de classe.

Pour spécifier le trafic qui ne doit pas correspondre à la carte de trafic, utilisez la commande **match not**. Par exemple, si la commande **match not** spécifie la chaîne « example.com », tout trafic qui inclut « example.com » ne correspond pas à la carte de trafic.

Pour le trafic que vous identifiez dans cette carte de trafic, vous spécifiez les actions à prendre sur le trafic dans la liste des politiques d'inspection.

Si vous souhaitez effectuer différentes actions pour chaque commande **match**, vous devez identifier le trafic directement dans la liste des politiques.

a) Créez la carte de trafic : **class-map type inspect im** [**match-all** | **match-any**] *class_map_name*

Où *nclass_map_name* est le nom de la carte de trafic. Le mot-clé **match-all** est le mot-clé par défaut. Il spécifie que le trafic doit correspondre à tous les critères pour correspondre à la carte de trafic. Le mot-clé match-any spécifie que le trafic correspond à la carte de trafic s'il correspond à au moins un énoncé **match**. L'interface de ligne de commande passe en mode de configuration class-map, où vous pouvez saisir une ou plusieurs commandes **match**.

b) (Facultatif) : Ajoutez une description à la carte de trafic : **description** *string*

Où *string* est la description de la carte de trafic (jusqu'à 200 caractères).

c) Précisez le trafic pour lequel vous souhaitez effectuer des actions à l'aide de l'une des commandes **match** suivantes. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

- **match** [**not**] **protocol {im-yahoo | im-msn}** : correspond à un protocole IM spécifique, Yahoo ou MSN.
- **match** [**not**] **service {chat | file-transfer** | **webcam** | **voice-chat** | **conference** | **games}** : correspond au service d'IM spécifique.
- **match** [**not**] **login-name regex** {*regex_name* | **class** *class_name*} : correspond au nom de connexion du client source du message IM en fonction de l'expression régulière ou de la liste d'expressions régulières spécifiée.
- **match** [**not**] **peer-login-name regex** {*regex_name* | **class** *class_name*} : correspond au nom de connexion de l'homologue de destination du message IM en fonction de l'expression régulière ou de la liste d'expressions régulières spécifiée.
- **match** [**not**] **ip-address** *ip_address mask*} : correspond à l'adresse IP source et au masque du message IM.
- **match** [**not**] **peer-ip-address** *ip_address mask*} : correspond à l'adresse IP de destination et au masque du message IM.
- **match** [**not**] **version regex** {*regex_name* | **class** *class_name*} : correspond à la version du message IM en fonction de l'expression régulière ou de la liste d'expressions régulières spécifiée.
- **match** [**not**] **filename regex** {*regex_name* | **class** *class_name*} : correspond au nom de fichier du message IM en fonction de l'expression régulière ou de la liste d'expressions régulières spécifiée. Cette correspondance n'est pas prise en charge pour le protocole MSN IM.

d) Saisissez **exit** pour quitter le mode de configuration de carte de trafic.

Étape 2 Créez une liste des politiques d'inspection IM : **policy-map type inspect im** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration de liste des politiques.

Étape 3 (Facultatif) Ajoutez une description à la liste des politiques : **description** *string*

Étape 4 Pour appliquer les actions au trafic correspondant, procédez comme suit.

a) Précisez le trafic sur lequel vous souhaitez effectuer des actions en utilisant l'une des méthodes suivantes :

- Si vous avez créé une carte de trafic IM, spécifiez-la en entrant la commande suivante : **class** *class_map_name*
- Précisez le trafic directement dans la liste des politiques à l'aide de l'une des commandes **match** décrites pour les cartes de classe IM. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

b) Précisez l'action que vous souhaitez effectuer sur le trafic correspondant en entrant la commande suivante :

- **drop-connection** [**log**] : abandonner le paquet et fermer la connexion.

- **reset** [**log**] : abandonne le paquet, ferme la connexion et envoie une réinitialisation TCP au serveur ou au client.
- **log** : envoie un message de journal système. Vous pouvez utiliser cette option seule ou avec l'une des autres actions.

Vous pouvez spécifier plusieurs commandes **class** ou **match** dans la liste des politiques. Pour en savoir plus sur l'ordre des commandes **class** et **match**, consultez Comment plusieurs classes de trafic sont gérées.

Exemple

L'exemple suivant montre comment définir une liste des politiques d'inspection IM.

```
hostname(config)# regex loginname1 "ying\@yahoo.com"
hostname(config)# regex loginname2 "Kevin\@yahoo.com"
hostname(config)# regex loginname3 "rahul\@yahoo.com"
hostname(config)# regex loginname4 "darshant\@yahoo.com"
hostname(config)# regex yahoo_version_regex "1\.0"
hostname(config)# regex gif_files ".*\.gif"
hostname(config)# regex exe_files ".*\.exe"

hostname(config)# class-map type regex match-any yahoo_src_login_name_regex
hostname(config-cmap)# match regex loginname1
hostname(config-cmap)# match regex loginname2

hostname(config)# class-map type regex match-any yahoo_dst_login_name_regex
hostname(config-cmap)# match regex loginname3
hostname(config-cmap)# match regex loginname4

hostname(config)# class-map type inspect im match-any yahoo_file_block_list
hostname(config-cmap)# match filename regex gif_files
hostname(config-cmap)# match filename regex exe_files

hostname(config)# class-map type inspect im match-all yahoo_im_policy
hostname(config-cmap)# match login-name regex class yahoo_src_login_name_regex
hostname(config-cmap)# match peer-login-name regex class yahoo_dst_login_name_regex

hostname(config)# class-map type inspect im match-all yahoo_im_policy2
hostname(config-cmap)# match version regex yahoo_version_regex

hostname(config)# class-map im_inspect_class_map
hostname(config-cmap)# match default-inspection-traffic

hostname(config)# policy-map type inspect im im_policy_all
hostname(config-pmap)# class yahoo_file_block_list
hostname(config-pmap-c)# match service file-transfer
hostname(config-pmap)# class yahoo_im_policy
hostname(config-pmap-c)# drop-connection
hostname(config-pmap)# class yahoo_im_policy2
hostname(config-pmap-c)# reset
hostname(config)# policy-map global_policy_name
hostname(config-pmap)# class im_inspect_class_map
hostname(config-pmap-c)# inspect im im_policy_all
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection des options IP

Vous pouvez configurer l'inspection des options IP pour contrôler les paquets IP qui sont autorisés en fonction du contenu du champ Options IP dans l'en-tête du paquet. Vous pouvez abandonner les paquets qui ont des options indésirables, effacer les options (et autoriser le paquet) ou autoriser le paquet sans modification.

Les options IP fournissent des fonctions de contrôle nécessaires dans certaines situations, mais inutiles pour la plupart des communications courantes. En particulier, les options IP comprennent des dispositions pour les horodatages, la sécurité et le routage spécial. L'utilisation des options IP est facultative, et le champ peut contenir zéro, une ou plusieurs options.

Pour obtenir la liste des options IP, avec des références aux RFC pertinents, consultez la page IANA, http://www.iana.org/assignments/ip-parameters/ip-parameters.xhtml.

L'inspection des options IP est activée par défaut, mais pour le trafic RSVP uniquement. Vous ne devez le configurer que si vous souhaitez autoriser des options supplémentaires que la liste par défaut, ou si vous souhaitez l'appliquer à d'autres types de trafic à l'aide d'une carte de classe de trafic d'inspection autre que celle par défaut.

Remarque L'inspection des options IP ne fonctionne pas sur les paquets fragmentés. Par exemple, les options ne sont pas effacées des fragments.

Les sections suivantes décrivent l'inspection des options IP.

Valeurs par défaut pour l'inspection des options IP

L'inspection des options IP est activée par défaut pour le trafic RSVP uniquement, à l'aide de la liste des politiques d'inspection _default_ip_options_map.

- L'option Router Alert (Alerte de routeur) est autorisée.

  Cette option informe les routeurs de transit d'inspecter le contenu du paquet même lorsque le paquet n'est pas destiné à ce routeur. Cette inspection est utile lors de la mise en œuvre de RSVP et de protocoles similaires qui nécessitent un traitement plutôt complexe de la part des routeurs le long du chemin de livraison du paquet. L'abandon de paquets RSVP contenant l'option Router Alert peut entraîner des problèmes dans les implémentations VoIP.

- Les paquets qui contiennent d'autres options sont abandonnés.

  Chaque fois qu'un paquet est abandonné en raison d'une inspection, le journal système 106 012 est émis. Le message indique quelle option a causé l'abandon. Utilisez la commande **show service-policy inspect ip-options** pour afficher les statistiques de chaque option.

Voici la configuration de la liste des politiques :

```
policy-map type inspect ip-options _default_ip_options_map
 description Default IP-OPTIONS policy-map
```

```
parameters
 router-alert action allow
```

Configurer une liste des politiques d'inspection des options IP

Si vous souhaitez effectuer une inspection d'options IP autres que celles par défaut, créez une carte de politiques d'inspection d'options IP pour préciser comment vous souhaitez gérer chaque type d'option.

Procédure

Étape 1 Créez une liste des politiques d'inspection des options IP : **policy-map type inspect ip-options** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration policy-map.

Étape 2 (Facultatif) Ajoutez une description à la liste des politiques : **description** *chaîne*

Étape 3 Entrez en mode de configuration des paramètres :

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

Étape 4 Déterminez les options que vous souhaitez autoriser.

Vous pouvez inspecter les options suivantes. Dans tous les cas, l'action **allow** autorise les paquets qui contiennent l'option sans modification; l'action **clear** autorise les paquets mais supprime l'option de l'en-tête.

Utilisez la forme **no** de la commande pour supprimer l'option de la liste. Tout paquet qui contient une option que vous n'incluez pas dans la liste est abandonné, même si le paquet contient des options autorisées ou effacées par ailleurs.

Pour obtenir la liste des options IP, avec des références aux RFC pertinents, consultez la page IANA, http://www.iana.org/assignments/ip-parameters/ip-parameters.xhtml.

- **default action** {**allow** | **clear**} : définit l'action par défaut pour toute option qui n'est pas explicitement incluse dans la liste. Si vous ne définissez pas d'action par défaut d'autorisation ou d'effacement, les paquets qui contiennent des options non autorisées sont abandonnés
- **basic-security action** {**allow** | **clear**} : autorise ou efface l'option Security (SEC).
- **commercial-security action** {**allow** | **clear**} : autorise ou efface l'option de sécurité commerciale (CIPSO).
- **eool action** {**allow** | **clear**} : autorise ou efface l'option de fin de liste d'options.
- **exp-flow-control action** {**allow** | **clear**} : autorise ou efface l'option de contrôle de flux expérimental (FINN).
- **exp-measurement action** {**allow** | **clear**} : autorise ou efface l'option de mesure expérimentale (ZSU).
- **extended-security action** {**allow** | **clear**} : autorise ou efface l'option de sécurité étendue (E-SEC).
- **imi-traffic-descriptor action** {**allow** | **clear**} : autorise ou efface l'option de descripteur de trafic IMI (IMITD).
- **nop action** {**allow** | **clear**} : autorise ou efface l'option Aucune opération.

- **quick-start action** {**allow** | **clear**} : autorise ou efface l'option de démarrage rapide (QS).
- **record-route action** {**allow** | **clear**} : autorise ou efface l'option de routage d'enregistrement (RR).
- **router-alert action** {**allow** | **clear**} : autorise ou efface l'option d'alerte de routeur (RTRALT).
- **timestamp action** {**allow** | **clear**} : autorise ou efface l'option d'horodatage (TS).
- {**0-255**} **action** {**allow** | **clear**} : autorise ou efface l'option identifiée par le numéro de type d'option. Le nombre est l'octet complet du type d'option (copie, carte et numéro d'option), et pas seulement la partie numéro d'option de l'octet. Ces types d'options peuvent ne pas représenter des options réelles. Les options non standard doivent être au format type-longueur attendu défini dans le protocole Internet RFC 791, http://tools.ietf.org/html/rfc791.

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection IPsec Pass Through

L'inspection IPsec Pass Through n'est pas activée dans la politique d'inspection par défaut. Vous devez donc l'activer si vous avez besoin de cette inspection. Cependant, la carte de trafic d'inspection par défaut comprend les ports IPsec par défaut, vous pouvez donc simplement modifier la politique d'inspection globale par défaut pour ajouter l'inspection IPsec. Vous pouvez également créer une nouvelle politique de service comme vous le souhaitez, par exemple, une politique spécifique à l'interface.

Les sections suivantes décrivent le moteur d'inspection IPsec Pass Through.

Aperçu de l'inspection de transmission directe IPsec

Internet Protocol Security (IPsec) est une suite de protocoles permettant de sécuriser les communications IP en authentifiant et en chiffrant chaque paquet IP d'un flux de données. IPsec comprend également des protocoles pour établir une authentification mutuelle entre les agents au début de la session et la négociation des clés cryptographiques à utiliser pendant la session. IPsec peut être utilisé pour protéger les flux de données entre une paire d'hôtes (par exemple, utilisateurs d'ordinateurs ou serveurs), entre une paire de passerelles de sécurité (comme les routeurs ou les pare-feu) ou entre une passerelle de sécurité et un hôte.

L'inspection applicative d'IPsec Pass Through (transmission directe IPsec) permet la traversée transparente du trafic ESP (protocole IP 50) et AH (protocole IP 51) associé à une connexion IKE sur le port UDP 500. Il évite une longue configuration d'ACL pour autoriser le trafic ESP et AH et offre également une sécurité à l'aide du délai d'expiration et du nombre maximal de connexions.

Configurez une liste des politiques pour la transmission directe IPsec afin de préciser les restrictions relatives au trafic ESP ou AH. Vous pouvez définir le nombre maximal de connexions par client et le délai d'inactivité.

Le trafic NAT et non NAT est autorisé. Cependant, la PAT n'est pas prise en charge.

Configurer une liste des politiques d'inspection de transmission directe IPsec

Une liste IPsec Passthrough vous permet de modifier les valeurs de configuration par défaut utilisées pour l'inspection de l'application IPsec Pass Through. Vous pouvez utiliser une liste IPsec Passthrough pour autoriser certains flux sans utiliser d'ACL.

La configuration comprend une carte par défaut, _default_ipsec_passthru_map, qui n'établit aucune limite maximale aux connexions ESP par client, et définit le délai d'inactivité ESP à 10 minutes. Vous ne devez configurer une carte de politiques d'inspection que si vous souhaitez des valeurs différentes ou si vous souhaitez définir des valeurs HA.

Procédure

Étape 1 Créer une liste des politiques d'inspection IPsec Pass Through : **policy-map type inspect ipsec-pass-thru** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration de liste des politiques.

Étape 2 (Facultatif) Ajoutez une description à la liste des politiques : **description** *string*

Étape 3 Pour configurer les paramètres qui influent sur le moteur d'inspection, procédez comme suit :

a) Entrez en mode de configuration des paramètres :

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b) Définissez un ou plusieurs paramètres. Vous pouvez définir les options suivantes; utilisez la forme **no** de la commande pour désactiver l'option :

- **esp per-client-max** *number* **timeout** *time* : autorise les tunnels ESP et définit le nombre maximal de connexions autorisées par client et le délai d'inactivité (au format hh:mm:ss). Pour autoriser un nombre illimité de connexions, spécifiez 0 pour le nombre.
- **ah per-client-max** *number* **délai d'expiration** *temps* : autorise les tunnels AH. Les paramètres ont la même signification que pour la commande esp.

Exemple

L'exemple suivant montre comment utiliser les listes de contrôle d'accès pour identifier le trafic IKE, définir une carte de paramètres IPsec Pass Thru, définir une politique et appliquer la politique à l'interface externe :

```
hostname(config)# access-list ipsecpassthruacl permit udp any any eq 500
hostname(config)# class-map ipsecpassthru-traffic
hostname(config-cmap)# match access-list ipsecpassthruacl
hostname(config)# policy-map type inspect ipsec-pass-thru iptmap
hostname(config-pmap)# parameters
hostname(config-pmap-p)# esp per-client-max 10 timeout 0:11:00
hostname(config-pmap-p)# ah per-client-max 5 timeout 0:06:00
```

```
hostname(config)# policy-map inspection_policy
hostname(config-pmap)# class ipsecpassthru-traffic
hostname(config-pmap-c)# inspect ipsec-pass-thru iptmap
hostname(config)# service-policy inspection_policy interface outside
```

Inspection IPv6

L'inspection IPv6 vous permet de consigner ou d'abandonner sélectivement le trafic IPv6 en fonction de l'en-tête d'extension. En outre, l'inspection IPv6 peut vérifier la conformité à la RFC 2460 pour le type et l'ordre des en-têtes d'extension dans les paquets IPv6.

L'inspection IPv6 n'est pas activée dans la politique d'inspection par défaut, vous devez donc l'activer si vous avez besoin de cette inspection. Vous pouvez simplement modifier la politique d'inspection globale par défaut pour ajouter l'inspection IPv6. Vous pouvez également créer une nouvelle politique de service comme vous le souhaitez, par exemple, une politique spécifique à l'interface.

Valeurs par défaut pour l'inspection IPv6

Si vous activez l'inspection IPv6 et que vous ne spécifiez pas de carte de politique d'inspection, la carte de politique d'inspection IPv6 par défaut est utilisée, et les actions suivantes sont effectuées :

- Autorise uniquement les en-têtes d'extension IPv6 connus. Les paquets non conformes sont abandonnés et journalisés.
- Applique l'ordre des en-têtes d'extension IPv6 tel que défini dans la spécification RFC 2460. Les paquets non conformes sont abandonnés et journalisés.
- Abandonne tout paquet avec un en-tête de type routage.

Voici la configuration de la liste des politiques :

```
policy-map type inspect ipv6 _default_ipv6_map
 description Default IPV6 policy-map
 parameters
  verify-header type
  verify-header order
 match header routing-type range 0 255
  drop log
```

Configurer une liste des politiques d'inspection IPv6

Pour identifier les en-têtes d'extension à abandonner ou à journaliser, ou pour désactiver la vérification des paquets, créez une liste des politiques d'inspection IPv6 à utiliser par la politique de service.

Procédure

Étape 1 Créer une liste des politiques d'inspection IPv6 : **policy-map type inspect ipv6** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration policy-map.

Étape 2 (Facultatif) Ajoutez une description à la liste des politiques : **description** *string*

Étape 3 (Facultatif) Abandonnez ou journalisez le trafic en fonction des en-têtes des messages IPv6.

a) Identifier le trafic en fonction de l'en-tête IPv6 : **match header** *type*

Où *type* est l'un des suivants :

- **ah** : correspond à l'en-tête de l'extension d'authentification IPv6.
- **count gt** *number* : spécifie le nombre maximal d'en-têtes d'extension IPv6, de 0 à 255.
- **destination-option** : correspond à l'en-tête d'extension IPv6 destination-option.
- **esp** : correspond à l'en-tête d'extension ESP (IPv6 Encapsulation Security Payload).
- **fragment** : correspond à l'en-tête d'extension fragment IPv6.
- **hop-by-hop** : correspond à l'en-tête d'extension IPv6 hop-by-hop.
- **routing-address count gt** *number* : définit le nombre maximal d'adresses d'en-tête de routage IPv6 de type 0, supérieur à un nombre entre 0 et 255.
- **routing-type** {**eq** | **range**} *number* : correspond au type d'en-tête de routage IPv6, de 0 à 255. Pour une plage, séparez les valeurs par un espace, par exemple, **30 40**.

b) Précisez l'action à effectuer sur les paquets correspondants. Vous pouvez abandonner le paquet et éventuellement le consigner, ou tout simplement le consigner. Si vous n'entrez pas d'action, le paquet est journalisé.

- **drop** [**log**] : abandon de tous les paquets qui correspondent.
- **log** : envoie un message de journal système. Vous pouvez utiliser cette option seule ou avec l'une des autres actions.

c) Répétez le processus jusqu'à ce que vous identifiiez tous les en-têtes que vous souhaitez abandonner ou journaliser.

Étape 4 Configurez les paramètres qui influent sur le moteur d'inspection.

a) Entrez dans le mode de configuration des paramètres.

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b) Définissez un ou plusieurs paramètres. Vous pouvez définir les options suivantes; utilisez la forme **no** de la commande pour désactiver l'option :

- **verify-header type** : autorise uniquement les en-têtes d'extension IPv6 connus.
- **verify-header order** : applique l'ordre des en-têtes d'extension IPv6 tel qu'il est défini dans la RFC 2460.

Exemple

L'exemple suivant crée une liste des politiques d'inspection qui supprimera et consignera tous les paquets IPv6 avec les en-têtes saut par saut, option de destination, adresse de routage et type de routage 0. Il applique également l'ordre et le type d'en-tête.

```
policy-map type inspect ipv6 ipv6-pm
 parameters
  verify-header type
  verify-header order
 match header hop-by-hop
  drop log
 match header destination-option
  drop log
 match header routing-address count gt 0
  drop log
 match header routing-type eq 0
  drop log

policy-map global_policy
 class class-default
  inspect ipv6 ipv6-pm
!
service-policy global_policy global
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection NetBIOS

L'inspection d'application NetBIOS effectue une NAT pour l'adresse IP intégrée dans les paquets du service de nom NetBIOS (NBNS) et les paquets des services de datagramme NetBIOS. Elle applique également la conformité du protocole en vérifiant la cohérence des champs de nombre et de longueur.

L'inspection NetBIOS est activée par défaut. Vous pouvez éventuellement créer une liste des politique abandonner ou journaliser les violations du protocole NetBIOS. La procédure suivante explique comment configurer une liste des politiques d'inspection NetBIOS.

Procédure

Étape 1 Créer une liste des politiques d'inspection NetBIOS : **policy-map type inspect netbios** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration policy-map.

Étape 2 (Facultatif) Ajouter une description à la liste des politiques : **description** *chaîne*

Étape 3 Entrez dans le mode de configuration des paramètres.

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

Étape 4 Précisez l'action à entreprendre pour les violations du protocole NETBIOS : **protocol-violation action** {**drop** [**log**] | **log**}

Où l'action **drop** abandonne le paquet. L'action **log** envoie un message de journal système lorsque cette liste des politiques correspond au trafic.

Exemple

```
hostname(config)# policy-map type inspect netbios netbios_map
hostname(config-pmap)# parameters
hostname(config-pmap-p)# protocol-violation drop log

hostname(config)# policy-map netbios_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# no inspect netbios
hostname(config-pmap-c)# inspect netbios netbios_map
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection PPTP

PPTP est un protocole de tunnellisation du trafic PPP. Une session PPTP est composée d'un canal TCP et généralement de deux tunnels GRE PPTP. Le canal TCP est le canal de contrôle utilisé pour négocier et gérer les tunnels PPTP GRE. Les tunnels GRE acheminent les sessions PPP entre les deux hôtes.

Lorsque cette option est activée, le moteur d'inspection PPTP inspecte les paquets du protocole PPTP et crée dynamiquement les connexions GRE et les xlates nécessaires pour autoriser le trafic PPTP.

Plus précisément, l'ASA inspecte les annonces de version PPTP et la séquence de demande/réponse d'appel sortant. Seul PPTP version 1, tel que défini dans la RFC 2637, est inspecté. L'inspection plus approfondie sur le canal de contrôle TCP est désactivée si la version annoncée par l'un ou l'autre des côtés n'est pas la version 1. En outre, la demande d'appel sortant et la séquence de réponse sont suivies. Les connexions et les xlates sont allouées dynamiquement selon les besoins pour permettre le trafic de données GRE secondaire subséquent.

Le moteur d'inspection PPTP doit être activé pour que le trafic PPTP soit traduit par PAT. En outre, la PAT n'est effectuée que pour une version modifiée de GRE (RFC2637) et uniquement si elle est négociée sur le canal de contrôle TCP PPTP. La PAT n'est pas effectuée pour la version non modifiée de GRE (RFC 1701 et RFC 1702).

Pour en savoir plus sur l'activation de l'inspection PPTP, consultez Configurer l'inspection du protocole de couche d'application.

Inspection RSH

L'inspection RSH est activée par défaut. Le protocole RSH utilise une connexion TCP du client RSH au serveur RSH sur le port TCP 514. Le client et le serveur négocient le numéro de port TCP où le client est à l'écoute du flux de sortie STDERR. L'inspection RSH prend en charge la NAT du numéro de port négocié, si nécessaire.

Pour en savoir plus sur l'activation de l'inspection RSH, consultez Configurer l'inspection du protocole de couche d'application.

Inspection SMTP et SMTP étendue

L'inspection ESMTP détecte les attaques, y compris les attaques de pourriels, d'hameçonnage, de messages malformés et les attaques de débordement ou d'insuffisance de la mémoire tampon. Elle prend également en charge la sécurité des applications et la conformité des protocoles, qui imposent la validité des messages ESMTP, bloquent les expéditeurs/récepteurs et bloquent le relais de messagerie.

L'inspection ESMTP est activée par défaut. Vous ne devez la configurer que si vous souhaitez un traitement différent de celui fourni par la carte d'inspection par défaut.

Les sections suivantes décrivent le moteur d'inspection ESMTP.

Aperçu de l'inspection SMTP et ESMTP

L'inspection des applications SMTP étendue (ESMTP) offre une protection améliorée contre les attaques basées sur SMTP en limitant les types de commandes SMTP qui peuvent passer par l'ASA et en ajoutant des fonctionnalités de surveillance. ESMTP est une amélioration du protocole SMTP et est similaire à SMTP à bien des égards.

L'inspection d'application ESMTP contrôle et réduit les commandes que l'utilisateur peut utiliser ainsi que les messages que le serveur renvoie. L'inspection ESMTP effectue trois tâches principales :

- Restreint les requêtes SMTP à sept commandes SMTP de base et à huit commandes étendues. Les commandes prises en charge sont les suivantes :
  - SMTP étendu : AUTH, EHLO, ETRN, HELP, SAML, SEND, SOML, STARTTLS et VRFY.
  - SMTP (RFC 821) : DATA, HELO, MAIL, NOOP, QUIT, RCPT, RSET.
- Surveille la séquence de commande-réponse SMTP.
- Génère une trace d'audit : l'enregistrement d'audit 108002 est généré lorsqu'un caractère non valide intégré dans l'adresse courriel est remplacé. Pour en savoir plus, consultez la RFC 821.

L'inspection ESMTP surveille la séquence de commande et de réponse pour les signatures anormales suivantes :

- Commandes tronquées.
- Fin de commande incorrecte (absence de terminaison par <CR><LR>).
- Les commandes MAIL et RCPT précisent qui sont l'expéditeur et le destinataire du courriel. Les adresses de messagerie sont analysées pour détecter les caractères inconnus. Le caractère de routage (|) est

supprimé (changé en espace vide) et « < » □« > » ne sont autorisés que s'ils sont utilisés pour définir une adresse de courriel (« > » doit être précédé de « < »).

- Transition inattendue du serveur SMTP.
- Pour les commandes inconnues ou non prises en charge, le moteur d'inspection modifie tous les caractères du paquet en X, qui sont rejetés par le serveur interne. Il en résulte un message tel que « 500 Command unknown: « XXX » (Commande inconnue : « XXX »). Les commandes incomplètes sont rejetées

  Les commandes ESMTP non prises en charge sont ATRN, ONEX, VERB, CHUNKING et les extensions privées.
- Modification du flux TCP.
- Pipelining de commandes.

Remarque Lorsque l'inspection ESMTP est activée, une session Telnet utilisée pour le protocole SMTP interactif peut se bloquer si les règles suivantes ne sont pas respectées : les commandes SMTP doivent comporter au moins quatre caractères ; elles doivent se terminer par un retour chariot et un saut de ligne; et vous devez attendre une réponse avant d'envoyer la commande suivante.

Valeurs par défaut pour l'inspection ESMTP

L'inspection ESMTP est activée par défaut, à l'aide de la liste des politiques d'inspection _default_esmtp_map.

- La bannière du serveur est masquée. Le moteur d'inspection ESMTP remplace les caractères de la bannière SMTP du serveur par des astérisques, à l'exception des caractères « 2 », « 0 », « 0 ». Les caractères de retour chariot (CR) et de retour de ligne (LF) sont ignorés.
- Les connexions chiffrées sont autorisées, mais pas inspectées.
- Les caractères spéciaux dans les adresses de l'expéditeur et du destinataire ne sont pas détectés et aucune action n'est entreprise.
- Les connexions dont la longueur de ligne de commande est supérieure à 512 sont abandonnées et journalisées.
- Les connexions de plus de 100 destinataires sont abandonnées et journalisées.
- Les messages dont la longueur du corps est supérieure à 998 octets sont journalisés.
- Les connexions dont la longueur de ligne d'en-tête est supérieure à 998 sont abandonnées et journalisées.
- Les messages avec des noms de fichier MIME supérieurs à 255 caractères sont abandonnés et journalisés.
- Les paramètres de réponse EHLO correspondant à « autres » sont masqués.

Voici la configuration de la liste des politiques :

```
policy-map type inspect esmtp _default_esmtp_map
 description Default ESMTP policy-map
 parameters
  mask-banner
  no mail-relay
```

```
 no special-character
 allow-tls
match cmd line length gt 512
 drop-connection log
match cmd RCPT count gt 100
 drop-connection log
match body line length gt 998
 log
match header line length gt 998
 drop-connection log
match sender-address length gt 320
 drop-connection log
match MIME filename length gt 255
 drop-connection log
match ehlo-reply-parameter others
 mask
```

Configurer une liste des politiques d'inspection ESMTP

Pour préciser les actions lorsqu'un message enfreint un paramètre, créez une liste des politiques d'inspection ESMTP. Vous pouvez ensuite appliquer la liste des politiques d'inspection lorsque vous activez l'inspection ESMTP.

Avant de commencer

Certaines options de correspondance de trafic utilisent des expressions régulières à des fins de correspondance. Si vous avez l'intention d'utiliser l'une de ces techniques, créez d'abord l'expression régulière ou la carte de classe d'expression régulière.

Procédure

Étape 1 Créez une liste des politiques d'inspection ESMTP : **policy-map type inspect esmtp** *policy_map_name*

Où *policy_map_name* est le nom de la liste des politiques. L'interface de ligne de commande passe en mode de configuration de la liste des politiques.

Étape 2 (Facultatif) Ajoutez une description à la liste des politiques : **description** *string*

Étape 3 Pour appliquer les actions au trafic correspondant, procédez comme suit.

a) Précisez le trafic pour lequel vous souhaitez effectuer des actions à l'aide de l'une des commandes **match** suivantes. Si vous utilisez une commande **match not**, l'action est appliquée à tout trafic qui ne correspond pas au critère dans la commande **match not**.

- **match** [**not**] **body** {**length** | **line length**} **gt** *bytes* : correspond aux messages dans lesquels la longueur ou la longueur d'une ligne dans un message de corps ESMTP est supérieure au nombre d'octets spécifié.
- **match** [**not**] **cmd Verb** *Verb1* [*Verb2*...] : correspond au verbe de commande dans le message. Vous pouvez spécifier une ou plusieurs des commandes suivantes : auth, data, ehlo, etrn, helo, help, mail, noop, quit, rcpt, rset, saml, soml, vrfy.
- **match** [**not**] **cmd line length gt** *bytes* : correspond aux messages dans lesquels la longueur d'une ligne dans le verbe de commande est supérieure au nombre d'octets spécifié.

- **match** [**not**] **cmd rcpt count gt** *count* : correspond aux messages dans lesquels le nombre de destinataires est supérieur au nombre spécifié.
- **match** [**not**] **ehlo-reply-parameter** *parameter* [*parameter2...*] :correspond aux paramètres de réponse EHLO ESMTP. Vous pouvez spécifier un ou plusieurs des paramètres suivants : 8bitmime, auth, binaryname, checkpoint, dsn, etrn, others, pipelining, size, vrfy.
- **match** [**not**] **header** {**length** | **line length**} **gt** *bytes* : correspond aux messages dans lesquels la longueur ou la longueur d'une ligne dans un en-tête ESMTP est supérieure au nombre d'octets spécifié.
- **match** [**not**] **header to-fields count gt** *count* : correspond aux messages dans lesquels le nombre de champs To dans l'en-tête est supérieur au nombre spécifié.
- **match** [**not**] **invalid-recipients count gt** *number* : correspond aux messages dans lesquels le nombre de destinataires non valides est supérieur au nombre spécifié.
- **match** [**not**] **mime filetype regex** {*regex_name* | **class** *class_name*} : correspond au type de fichier MIME ou multimédia avec l'expression régulière ou la liste d'expressions régulières spécifiée.
- **match** [**not**] **mime filename length gt** *bytes* : correspond aux messages dans lesquels le nom de fichier est plus long que le nombre d'octets spécifié.
- **match** [**not**] **mime encoding** *type* [*type2...*] : correspond au type d'encodage MIME. Vous pouvez spécifier un ou plusieurs des types suivants : 7bit, 8bit, base64, binary, others, quoted-printable.
- **match** [**not**] **sender-address regex** {*regex_name* | **class** *class_name*} : fait correspondre l'adresse courriel de l'expéditeur à l'expression régulière ou à la liste d'expressions régulières spécifiée.
- **match** [**not**] **sender-address length gt** *bytes* : correspond aux messages dans lesquels l'adresse de l'expéditeur est supérieure au nombre d'octets spécifié.

b) Précisez l'action que vous souhaitez effectuer sur le trafic correspondant en saisissant l'une des commandes suivantes :

- **drop-connection** [**log**] : abandonner le paquet et fermer la connexion.
- **mask** [**log**] : masque la partie correspondante du paquet. Cette action est disponible uniquement pour **ehlo-reply-parameter** et **cmd verb**.
- **reset** [**log**] : abandonne le paquet, ferme la connexion et envoie une réinitialisation TCP au serveur ou au client.
- **log** : envoie un message de journal système. Vous pouvez utiliser cette option seule ou avec l'une des autres actions.
- **rate-limit** *message_rate* : limite le débit de messages en paquets par seconde. Cette option est disponible uniquement avec **cmd verb**, où vous pouvez l'utiliser comme seule action ou vous pouvez l'utiliser en conjonction avec l'action **mask**.

Vous pouvez spécifier plusieurs commandes **match** dans la liste des politiques. Pour en savoir plus sur l'ordre des commandes **match**, consultez Comment plusieurs classes de trafic sont gérées.

Étape 4 Pour configurer les paramètres qui influent sur le moteur d'inspection, procédez comme suit :

a) Entrez en mode de configuration des paramètres :

```
hostname(config-pmap)# parameters
hostname(config-pmap-p)#
```

b) Définissez un ou plusieurs paramètres. Vous pouvez définir les options suivantes; utilisez la forme **no** de la commande pour désactiver l'option :

- **mail-relay** *domain-name* **action** {**drop-connection** [**log**] | **log**} : identifie un nom de domaine pour le relais de messagerie. Vous pouvez soit abandonner la connexion et éventuellement la journaliser, soit la journaliser.
- **mask-banner** : masque la bannière du serveur ESMTP.
- **special-character action** {**drop-connection** [**log**] | **log**} : identifie l'action à prendre pour les messages qui comprennent les caractères spéciaux pipe (|), guillemet arrière et NUL dans les adresses courriel de l'expéditeur ou du destinataire. Vous pouvez soit abandonner la connexion et éventuellement la journaliser, soit la journaliser.
- **allow-tls** [**action log**] : indique s'il faut autoriser ESMTP sur TLS (connexions chiffrées) sans inspection. Vous pouvez éventuellement journaliser les connexions chiffrées. La valeur par défaut est d'autoriser les sessions TLS sans inspection. Si vous spécifiez **no allow-tls**, le système supprime l'indication STARTTLS de la connexion de session et force une connexion en texte brut.

Exemple

L'exemple suivant montre comment définir une liste des politiques d'inspection ESMTP.

```
hostname(config)# regex user1 "user1@cisco.com"
hostname(config)# regex user2 "user2@cisco.com"
hostname(config)# regex user3 "user3@cisco.com"
hostname(config)# class-map type regex senders_black_list
hostname(config-cmap)# description "Regular expressions to filter out undesired senders"
hostname(config-cmap)# match regex user1
hostname(config-cmap)# match regex user2
hostname(config-cmap)# match regex user3

hostname(config)# policy-map type inspect esmtp advanced_esmtp_map
hostname(config-pmap)# match sender-address regex class senders_black_list
hostname(config-pmap-c)# drop-connection log

hostname(config)# policy-map outside_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect esmtp advanced_esmtp_map

hostname(config)# service-policy outside_policy interface outside
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection SNMP

L'inspection des applications SNMP est appliquée à la fois au trafic vers le périphérique et au trafic traversant le périphérique. Cette inspection est nécessaire si vous configurez SNMP v3 où les utilisateurs sont limités à des hôtes SNMP spécifiques. Sans inspection, un utilisateur v3 défini peut interroger le périphérique à partir de n'importe quel hôte autorisé. L'inspection SNMP est activée par défaut pour les ports par défaut. Vous ne devez donc la configurer que si vous utilisez des ports autres que les ports par défaut. Les ports par défaut sont UDP/161, 162 (pour tous les types de périphériques) et UDP/4161 pour les périphériques qui exécutent également FXOS, car FXOS écoute sur UDP/161.

Par défaut, l'inspection SNMP limite l'interrogation à la version configurée.

Remarque Si vous configurez SNMP sur un périphérique qui exécute également FXOS, l'inspection SNMP est obligatoire et est réactivée si vous la désactivez. L'inspection SNMP est activée sur une classe de trafic qui comprend le port UDP/4161.

Vous pouvez également restreindre le trafic SNMP à une version particulière de SNMP. Les versions antérieures de SNMP sont moins sécurisées; par conséquent, le refus de certaines versions SNMP peut être requis par votre politique de sécurité. Le système peut refuser les versions SNMP 1, 2, 2c ou 3. Vous contrôlez les versions autorisées en créant une liste SNMP, comme expliqué ci-dessous. Si vous n'avez pas besoin de contrôler les versions, activez simplement l'inspection SNMP sans liste.

Procédure

Créez une liste SNMP.

Utilisez la commande **snmp-map** *map_name* pour créer la liste et passer en mode de configuration de liste SNMP, puis la commande **deny version** *version* pour identifier les versions à refuser. La version peut être 1, 2, 2c ou 3.

Exemple :

L'exemple suivant refuse les versions SNMP 1 et 2 :

```
hostname(config)# snmp-map sample_map
hostname(config-snmp-map)# deny version 1
hostname(config-snmp-map)# deny version 2
```

Prochaine étape

Vous pouvez maintenant configurer une politique d'inspection pour utiliser la liste. Consultez Configurer l'inspection du protocole de couche d'application.

Inspection SQL*Net

L'inspection SQL*Net est activée par défaut. Le moteur d'inspection prend en charge les versions 1 et 2 de SQL*Net, mais uniquement le format Transparent Network Substrate (TNS). L'inspection ne prend pas en charge le format Tabular Data Stream (TDS). Les messages SQL*Net sont analysés pour rechercher les adresses et les ports intégrés, et la réécriture NAT est appliquée au besoin.

L'affectation de port par défaut pour SQL*Net est 1521. Il s'agit de la valeur utilisée par Oracle pour SQL*Net, mais cette valeur n'est pas conforme aux affectations de port IANA pour Structured Query Language (SQL). Si votre application utilise un port différent, appliquez l'inspection SQL*Net à une carte de trafic qui comprend ce port.

Désactiver l'inspection SQL*Net dans les cas suivants :

- Le transfert de données SQL se produit sur le même port que le port TCP de contrôle SQL 1521. Le périphérique de sécurité agit comme un serveur proxy lorsque l'inspection SQL*Net est activée et réduit la taille de la fenêtre client de 65 000 à environ 16 000, ce qui entraîne des problèmes de transfert de données.
- L'inspection de débits élevés de trafic SQL entraîne des pointes inacceptables dans l'utilisation du processeur.

Après avoir désactivé l'inspection SQL*Net, utilisez la commande **clear conn port 1521** pour que les connexions puissent être recréées sans inspection.

Pour en savoir plus sur l'activation de l'inspection SQL*Net, consultez Configurer l'inspection du protocole de couche d'application.

Inspection Sun RPC

Cette section décrit l'inspection des applications Sun RPC.

Aperçu de l'inspection Sun RPC

L'inspection du protocole Sun RPC est activée par défaut. Vous devez simplement gérer la table du serveur Sun RPC pour identifier les services autorisés à traverser le pare-feu. Cependant, l'ouverture dynamique de ports pour NFS est effectuée pour n'importe quel serveur, même sans configuration de la table du serveur.

Sun RPC est utilisé par NFS et NIS. Les services Sun RPC peuvent s'exécuter sur n'importe quel port. Lorsqu'un client tente d'accéder à un service Sun RPC sur un serveur, il doit connaître le port sur lequel le service est exécuté. Il le fait en interrogeant le processus de mappage de port, généralement rpcbind, sur le port bien connu de 111.

Le client envoie le numéro de programme Sun RPC du service et le processus de mappage de port répond en indiquant le numéro de port du service. Le client envoie ses requêtes Sun RPC au serveur, en précisant le port identifié par le processus de mappage de port. Lorsque le serveur répond, l'ASA intercepte ce paquet et ouvre les connexions TCP et UDP embryonnaires sur ce port.

La NAT ou la PAT des informations de charge utile Sun RPC n'est pas prise en charge.

Gérer les services Sun RPC

Utilisez la table des services Sun RPC pour contrôler le trafic de RPC en fonction des sessions de RPC établies.

Procédure

Étape 1 Configurez les propriétés du service Sun RPC.

sunrpc-server *interface_name ip_address mask* **service** *service_type* **protocol** {**tcp** | **udp**} *port*[*-port*] **timeout** *hh:mm:ss*

Lieu :

- *interface_name* : l'interface par laquelle le trafic passe vers le serveur.
- *ip_address mask* : l'adresse du serveur Sun RPC.
- **service** *service_type* : le type de service sur le serveur, qui est le mappage entre un type de service spécifique et le numéro de port utilisé pour le service. Pour déterminer le type de service (par exemple, 100003), utilisez la commande **sunrpcinfo** dans la ligne de commande UNIX ou Linux sur la machine du serveur Sun RPC.
- **protocol** {**tcp** | **udp**} : si le service utilise TCP ou UDP.
- *port*[*-port*] : le port ou la plage de ports utilisés par le service. Pour spécifier une plage de ports, séparez les numéros de ports de début et de fin dans la plage par un tiret (par exemple, 111-113).
- **timeout** *hh:mm:ss* : délai d'inactivité du passage ouvert pour la connexion par l'inspection Sun RPC.

Exemple :

Par exemple, pour créer un délai d'expiration de 30 minutes pour le serveur Sun RPC avec l'adresse IP 192.168.100.2, saisissez la commande suivante. Dans cet exemple, le serveur Sun RPC se trouve sur l'interface interne en utilisant le port TCP 111.

```
hostname(config)# sunrpc-server inside 192.168.100.2 255.255.255.255
service 100003 protocol tcp 111 timeout 00:30:00
```

Étape 2 (Facultatif) Surveillez les passages créés pour ces services.

Pour afficher les passages ouverts pour les services Sun RPC, saisissez la commande **show sunrpc-server active**. Par exemple :

```
hostname# show sunrpc-server active
LOCAL FOREIGN SERVICE TIMEOUT
-----------------------------------------------
1 209.165.200.5/0 192.168.100.2/2049 100003 0:30:00
2 209.165.200.5/0 192.168.100.2/2049 100003 0:30:00
3 209.165.200.5/0 192.168.100.2/647 100005 0:30:00
4 209.165.200.5/0 192.168.100.2/650 100005 0:30:00
```

L'entrée dans la colonne LOCAL affiche l'adresse IP du client ou du serveur sur l'interface interne, tandis que la valeur dans la colonne FOREIGN affiche l'adresse IP du client ou du serveur sur l'interface externe.

Au besoin, vous pouvez effacer ces services à l'aide de **clear sunrpc-server active**

Inspection TFTP

L'inspection TFTP est activée par défaut.

TFTP, décrit dans la RFC 1350, est un protocole simple pour lire et écrire des fichiers entre un serveur TFTP et un client.

Le moteur d'inspection inspecte la demande de lecture (RRQ), la demande d'écriture (WRQ) et la notification d'erreur (ERROR) et crée dynamiquement des connexions et des traductions, au besoin, pour permettre le transfert de fichiers entre un client et un serveur TFTP.

Un canal secondaire dynamique et une traduction de PAT, le cas échéant, sont alloués lors de la réception d'une demande valide de lecture (RRQ) ou d'écriture (WRQ). Ce canal secondaire est ensuite utilisé par TFTP pour le transfert de fichiers ou la notification d'erreurs.

Seul le serveur TFTP peut initier le trafic sur le canal secondaire, et au plus un canal secondaire incomplet peut exister entre le client et le serveur TFTP. Une notification d'erreur du serveur ferme le canal secondaire.

L'inspection TFTP doit être activée si la PAT statique est utilisée pour rediriger le trafic TFTP.

Pour en savoir plus sur l'activation de l'inspection TFTP, consultez Configurer l'inspection du protocole de couche d'application.

Inspection XDMCP

XDMCP est un protocole qui utilise le port UDP 177 pour négocier les sessions X, qui utilisent TCP lorsqu'elles sont établies.

Pour une négociation réussie et le démarrage d'une session XWindows, l'ASA doit autoriser la connexion TCP de retour à partir de l'ordinateur hébergeant X. Pour autoriser la connexion de retour, vous pouvez utiliser des règles d'accès pour autoriser les ports TCP. Sinon, vous pouvez exécuter la commande **established** sur l'ASA. Une fois que XDMCP a négocié le port pour envoyer l'affichage, la commande **established** est consultée pour vérifier si cette connexion arrière doit être autorisée.

Pendant la session XWindows, le gestionnaire communique avec le serveur X d'affichage sur le port bien connu 6000 | n. Chaque affichage a une connexion distincte au serveur X, en raison du paramètre de terminal suivant.

```
setenv DISPLAY Xserver:n
```

où *n* est le numéro d'affichage.

Lorsque XDMCP est utilisé, l'affichage est négocié à l'aide d'adresses IP, sur lesquelles l'ASA peut appliquer la NAT si nécessaire. L'inspection XDMCP ne prend pas en charge la PAT.

Pour en savoir plus sur l'activation de l'inspection XDMCP, consultez Configurer l'inspection du protocole de couche d'application.

Inspection VXLAN

L'inspection du réseau local virtuel extensible (VXLAN) fonctionne sur le trafic encapsulé VXLAN qui passe par l'ASA. Il garantit que le format d'en-tête VXLAN est conforme aux normes, en abandonnant les paquets malformés. L'inspection VXLAN n'est pas effectuée sur le trafic pour lequel l'ASA agit comme point d'extrémité de tunnel VXLAN (VTEP) ou passerelle VXLAN, car ces vérifications sont effectuées dans le cadre normal de la désencapsulation des paquets VXLAN.

Les paquets VXLAN sont UDP, normalement sur le port 4789. Ce port fait partie de la classe par défaut-inspection-trafic, vous pouvez donc simplement ajouter l'inspection VXLAN à la règle de politique de service inspection_default. Vous pouvez également créer une carte de trafic pour celui-ci en utilisant la mise en correspondance de port ou d'ACL.

Historique de l'inspection du protocole Internet de base

Nom de la caractéristique	Versions	Renseignements sur les fonctionnalités
Prise en charge de l'inspection DCERPC pour le message UUID ISystemMapper, RemoteGetClassObject opnum3.	9.4(1)	L'ASA a commencé à prendre en charge les messages DCERPC non EPM dans la version 8.3, en prenant en charge le message UUID ISystemMapper, RemoteCreateInstance opnum4. Cette modification étend la prise en charge au message RemoteGetClassObject opnum3. Nous n'avons pas modifié de commandes.
Inspection des paquets VXLAN	9.4(1)	L'ASA peut inspecter l'en-tête VXLAN pour faire respecter le format standard. Nous avons introduit la commande suivante : **inspect vxlan**.
L'inspection ESMTP change de comportement par défaut pour les sessions TLS.	9.4(1)	La valeur par défaut pour l'inspection ESMTP a été modifiée pour autoriser les sessions TLS, qui ne sont pas inspectées. Cependant, cette valeur par défaut s'applique aux systèmes nouveaux ou recréés. Si vous mettez à niveau un système qui inclut **no allow-tls**, la commande n'est pas modifiée. Le changement de comportement par défaut a également été apporté dans ces anciennes versions : 8.4(7.25), 8.5(1.23), 8.6(1.16), 8.7(1.15), 9.0(4.28), 9.1(6.1), 9.2(3.2) 9.3(1.2), 9.3(2.2).

Nom de la caractéristique	Versions	Renseignements sur les fonctionnalités
Améliorations de l'inspection des options d'IP	9.5(1)	L'inspection des options IP prend désormais en charge toutes les options IP possibles. Vous pouvez régler l'inspection pour autoriser, effacer ou abandonner toutes les options standard ou expérimentales, y compris celles non encore définies. Vous pouvez également définir un comportement par défaut pour les options non définies explicitement dans une carte d'inspection des options IP. Nous avons ajouté les commandes suivantes : **basic-security**, **commercial-security**, **default**, **exp-flow-control**, **exp-measure**, **extended-security**, **imi-traffic-description**, **quick-start**, **record-route**, **timestamp**, et {**0-255**} (qui indique un numéro de type d'option IP).
Améliorations de l'inspection DCERPC et du filtrage des UUID	9.5(2)	L'inspection DCERPC prend désormais en charge la NAT pour les messages opnum5 d'OxidResolver ServerAlive2. Vous pouvez également filtrer les identifiants universels (UUID) des messages DCERPC pour réinitialiser ou enregistrer des types de messages particuliers. Il existe une nouvelle carte des classes d'inspection DCERPC pour le filtrage des UUID. Nous avons introduit la commande suivante : **match** [**not**] **uuid**. Nous avons modifié la commande suivante : **class-map type inspect**.
Inspection DNS sur TCP	9.6(2)	Vous pouvez désormais inspecter le trafic DNS sur TCP (TCP/53). Nous avons ajouté la commande suivante : **tcp-inspection**.
Prise en charge de Cisco Umbrella	9.10(1)	Vous pouvez configurer le périphérique pour rediriger les requêtes DNS vers Cisco Umbrella, afin que votre politique de sécurité d'entreprise définie dans Cisco Umbrella puisse être appliquée aux connexions des utilisateurs. Vous pouvez autoriser ou bloquer les connexions en fonction du FQDN ou, pour les FQDN suspects, vous pouvez rediriger l'utilisateur vers le mandataire intelligent Cisco Umbrella, qui peut effectuer le filtrage d'URL. La configuration de Cisco Umbrella fait partie de la politique d'inspection DNS. Nous avons ajouté ou modifié les commandes suivantes : **umbrella** (modes de configuration des paramètres globaux et de liste des politiques), **token**, **public-key**, **timeout edns**, **dnscrypt**, **show service-policy inspect dns detail**.

Nom de la caractéristique	Versions	Renseignements sur les fonctionnalités
Améliorations de Cisco Umbrella.	9.12(1)	Vous pouvez maintenant identifier les noms de domaine locaux qui doivent contourner Cisco Umbrella. Les demandes DNS pour ces domaines sont directement transmises aux serveurs DNS sans passer par Cisco Umbrella. Vous pouvez également identifier les serveurs Cisco Umbrella à utiliser pour la résolution des demandes DNS. Enfin, vous pouvez définir la politique d'inspection de Cisco Umbrella pour qu'elle s'ouvre même en cas de non-conformité, afin que les demandes DNS ne soient pas bloquées si le serveur Cisco Umbrella n'est pas disponible. Nous avons ajouté ou modifié les commandes suivantes : **local-domain-bypass**, **resolver**, **umbrella fail-open**.
Inspection XDMCP désactivée par défaut dans les nouvelles installations.	9.15(1)	Auparavant, l'inspection XDMCP était activée par défaut pour tout le trafic. Désormais, sur les nouvelles installations, qui comprennent les nouveaux systèmes et les systèmes recréés, XDMCP est désactivé par défaut. Si vous avez besoin de cette inspection, veuillez l'activer. Notez que lors des mises à niveau, vos paramètres actuels pour l'inspection XDMCP sont conservés, même si vous les aviez simplement activés via les paramètres d'inspection par défaut.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d’incohérence, la version anglaise du présent contenu prévaudra.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d’incohérence, la version anglaise du présent contenu prévaudra.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d’incohérence, la version anglaise du présent contenu prévaudra.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d’incohérence, la version anglaise du présent contenu prévaudra.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d’incohérence, la version anglaise du présent contenu prévaudra.